Trading 1Min Bar Russell Futures Using
The Fading Memory Polynomial Velocity Strategy

August 12 2011 –August 12 2016

Working Paper August 2016

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.

This is a mathematical technique that fits a n^{th} order polynomial to the last N price bars but calculates the coefficients of the polynomial such that the error between the current n^{th} order polynomial and the current bar is weighted much higher than the error between the price N bars ago and the value of the n^{th} order polynomial N bars ago. As an example, if the latest price is at time t and the price made a turn at time bar t-10, then we do not want prices prior to t-10 affecting the current polynomial fit as much. As will be shown the most familiar case of this fading memory technique is the exponential moving average. The fading memory technique is in contrast to the Least Squares Polynomial fit, which weights all past errors between the polynomial and the price bar equally.

In previous working papers at **http://www.meyersanalytics.com/papers.html** we showed how the application of a price curve generated by Nth Order Fixed Memory Polynomial Velocity could be used to develop a strategy to buy and sell futures intraday. The reasoning behind this type of strategy was to only trade when the price trend velocity was above a certain threshold. Many times prices meander around without any notable trend and this is considered noise. During these times we do not wish to trade because of the cost of whipsaw losses that would occur from this type of price action. When a price trend finally starts, the velocity of that price trend moves above a minimum threshold noise value. Thus the velocity strategy would only issue a trade when certain velocity thresholds above "noise" levels are crossed.

The velocity strategy that we will use here to trade the Mini Russell futures contract is called the Nth Order ***Fading*** Memory Adaptive Polynomial Velocity Strategy. The word "Adaptive" is used because the polynomial inputs change over time, adapting to the changing trading patterns of the Russell futures contract. The Nth Order Fading memory Adaptive Polynomial Velocity Strategy has a number of unknown inputs that we have to determine before we can use this strategy to trade. These unknown inputs to the Fading Memory Polynomial are the polynomial order(degree), the optimum number of prices we need to determine the coefficients of the polynomial and finally the velocity thresholds. Here we will use Walk Forward Optimization and out-of-sample performance to determine the "best" polynomial inputs as well as how these inputs should change over time. We will use the nth Order Fading Memory Adaptive Polynomial Velocity Strategy to trade the Mini Russell futures contract on an intraday basis using one minute bar price data. To test this strategy we will use one minute bar prices of the Russell futures contract(TF) traded on the Intercontinental Exchange for the five years from August 12, 2011 to August 12, 2016.

The n^{th} Order Fading Memory Adaptive Polynomial Velocity Defined

The adaptive n^{th} order Fading Memory Polynomial Velocity is constructed and plotted at each bar by solving for the coefficients $\mathbf{b_1, b_2, b_3, \ldots b_n}$ for the discrete orthogonal Meixner polynomials at each bar using the exponential decay factor $\boldsymbol{\alpha=(1-\beta)}$ and the equation for $\mathbf{b_j}$ shown in the **"Math"** appendix of this paper. Then **Velocity(t+1)** is constructed from the equation shown in the "Math" appendix and plotted under the price chart.

The velocity of a 2^{nd}, 3^{rd} and 4^{th} order polynomial should change faster than the straight line (1^{st} order velocity). As observed from the 2^{nd} order velocity equation in the "Math" section, there is an acceleration component in the calculation of the velocity. This means that the 2^{nd} order velocity will reflect a change in the price trend much faster than the straight line velocity which does not have an acceleration component. The same is true for 3^{rd} and 4^{th} order velocities. Whether higher order polynomial velocities is an advantage or not we will let the computer decide when we let the computer search for the "best" polynomial degree as described below.

At each bar we calculate the nth order (1st through 4th) fading memory polynomial velocity from the formulas in the "Math" appendix. As we will show below, optimization will determine the order or degree for nth order polynomial velocity that will be used. 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.

The strategy follows the velocity curve. When the velocity is greater than the threshold amount ***vup*** a buy signal is issued. The threshold **vup** serves as a noise filter. That is, price noise creates a lot of small back and forth velocity movement. Unless the velocity can break some threshold to the upside, no trade is issued and the move is considered price noise. The same logic holds for the sell threshold **vdn**.

Intraday Bars Exit Rule:
Close the position at 1500 CST when the U.S stock market trading stops. (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 after the opening until 9:00am CST

To test this strategy we will use one minute bar prices of the mini Russell 2000 index futures contract traded on the Intercontinental Exchange (ICE) and known by the symbol TF for the 282 trading weeks from August 12, 2011 to August 12, 2016.

We will test the Fading Memory strategy with the above TS 1 min bars on a ***walk forward basis***, where the in-sample(IS) will be 30 calendar days and the out-of-sample(OOS) will be the next 7 calendar days following as will be described below. The 7 calendar day OOS periods will end on a Friday as will the 30 calendar day IS periods.

Testing The Polynomial Velocity Strategy Using Walk Forward Optimization

There will be four strategy parameters to determine:

1. **degree**, degree=1 for straight line velocity, degree=2 for 2nd order velocity, etc.
2. **alpha =(1-β)** The exponential decay weight for the Nth Order Fading Memory Polynomial calculation**.**
3. ***vup***, the threshold amount that velocity has to be greater than to issue a buy signal
4. ***vdn***, the threshold amount that velocity has to be less than to issue a sell signal

We will test this strategy with the above TF 1 min bars on a walk forward basis, as will be described below. To create our walk forward files we will use the ***add-in*** software product called the Power Walk Forward Optimizer (PWFO). In TradeStation (TS), we will run the PWFO strategy ***add-in*** along with the nth Order Fading Memory Polynomial Velocity Strategy on the TF 1min data from July 7, 2011 to August 12, 2016. The PWFO will breakup and create 30 day calendar in-sample sections along with their corresponding one calendar week out-of-sample sections from the 267 weeks of TF (see Walk forward Testing below) creating 262 out-of-sample weeks.

What Is An In-Sample Section and Out-Of-Sample Section?

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 an ***in-sample*** or ***test 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" (see Walk Forward Out-of-Sample Testing section below) it is important to see how the strategy inputs chosen from the in-sample section perform on out-of-sample data.

What Does The Power Walk Forward Optimizer (PWFO) Do?

The PWFO is a TS ***add-in*** that breaks up the TS optimization run into a number of user selectable in-sample and out-of-sample sections. The PWFO prints out the in-sample sample performance metrics **and the out-of-sample performance results**, on one line, for each case or strategy input variable combination that is run by the TradeStation(TS) optimization module to a user selected spreadsheet comma delimited file. The PWFO can generate up to 500 different in-sample and out-of-sample date optimization files in one TS run, saving the user from having to generate optimization runs one at a time. The PWFO output allows you to quickly determine whether your procedure for selecting input parameters for your strategy just curve fits the price and noise, or produces statistically valid out-of-sample results. In addition to the out-of-sample performance results presented for each case, 30+ superior and robust performance metrics (many are new and never presented before) are added to each case line in the in-sample section and printed out to the comma delimited file. These 30+ performance metrics allow for a superior and robust selection of input variables from the in-sample section that have a higher probability of performing well on out-of-sample data (Please see http://meyersanalytics.com/Walk-Forward-Optimization.html for a listing of these performance metrics).

For our computer run we will have the PWFO breakup the 262 weeks of TF one minute bar price data into 262 in-sample and out-of sample files. The in-sample sections will be 30 calendar days and the out-of-sample(oos) section will be the one week following the in-sample section. The oos week will always end on a Friday as will the 30 day calendar in-sample section. As an example the first in-sample section would be from 7/7/2011 to 8/5/2011 and the out-of-sample section would be the week following from 8/8/2011 to 8/12/2011.(our in-sample and out-of-sample sections always end on a Friday). We would then move everything ahead a week and the 2nd in-sample section would be from 7/14/2011 to 8/12/2011 and the week following out-of-sample section would be from 8/15/2011 to 8/19/2011. Etc.

The PWFO 262 in-sample/out-of-sample section dates are shown in **Table 1** below. We will then use another software product called the Walk Forward Performance Metric Explorer (WFME) on each of the 262 in-sample and out-of-sample(oos) sections generated by the PWFO to find the best in-sample section performance ***filter*** that determines the strategy input parameters ***(degree, N, vup, vdn)*** that will be used on the out-of-sample data. Detailed information about the PWFO and the WFME can be found at www.meyersanalytics.com

For the in-sample data we will run the TradeStation optimization engine on the 262 weeks of TF 1 min bars with the following ranges for the nth order fading memory polynomial velocity strategy input variables.

1. **pw=degree from 1 to 4**
2. **N from 20 to 70 in steps of 10.**
3. **vup from 0.25 to 3.5 steps of 0.25**
4. **vdn from 0.25 to 3.5 in steps of 0.25**
5. **Mult = 1, iNorm=1 (See Appendix 2, the Normalization Multiplier)**

Note: I use **N** because it gives a better understanding of how many bars of past data are approximately being used. **N** and α **(α=1-β)** are approximately related by the formula **α=2/(1+N)**. N is converted to α by this formula in the Nth Order Fading Memory Polynomial calculation

This will produce 4704 different cases or combinations of the input parameters for each of the 262 PWFO output files.

Walk Forward Out-of-Sample Testing

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 in-sample 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 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 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, then we will have an estimate of how our strategy will perform on average. Due to the Central Limit Theorem, as our 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"***? 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 input parameter combinations, the best performance parameters will be from those strategy 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 movements that were captured by a certain set of input parameters were a large part of the total net profits, as they usually are in real price movements, then choosing these input parameters will produce losses when traded on future data. These losses occur because the spurious movements will not be repeated in the same way. This is why strategy optimization, neural net optimizations or combinatorial searches with no out-of-sample testing cause loses when traded in real time by mistaking chance fluctuations for genuine effects. Unfortunately it is human nature to extrapolate past performance to project future trading results and thus results from the chance fluctuations of 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 three 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 in-sample/oos sections and take an average of our weekly results over all out-of-sample sections. This average gives us an expected weekly return and a standard deviation of weekly returns which allows us to statistically estimate the expected equity and its path ranges for N weeks in the future.

Finding The Strategy Input Parameters in The Walk Forward Test Sections

The PWFO generates a number of performance metrics in the in-sample section. (Please see **http://meyersanalytics.com/Walk-Forward-Optimization.html** for a listing of these performance metrics).The question we are attempting to answer statistically, is which performance metric or combination of performance metrics (which we will call a ***filter***) in 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 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 out-of-sample sections. The PWFO produces a total of 32 different performance metrics in the in-sample section. If we have 4704 different input variations or cases then the in-sample section consists of 32 columns of performance metrics for each of the 4704 different strategy inputs or rows.

An example of a simple filter would be to choose the row in the in-sample section that had the highest net profit or perhaps a row that had one the best performance metric from one of the other 32 PWFO metrics. Unfortunately it was found that this type of simple 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 a combination ***filter*** that is used in this paper with good out-of-sample results. High profit factors (**PF**) in the in-sample section usually mean poor performance in the out-of-sample-section. This is a kind of reversion to the mean. So in the in-sample section we eliminate all strategy input rows that have a **PF>4** . The PWFO generates the metric **eqTrn.** This metric is the **Slope Of The Straight Line Fit ToThe Trade Equity Line in the In-Sample Section**. Let us choose the 50 rows in the in-sample section that contain the highest(top) slope of the trade equity fitted straight line from the rows that are left from the **PF** elimination. This particular filter will now leave 50 cases or rows in the in-sample section that satisfy the above filter conditions. Suppose for this filter, within the 50 in-sample rows that are left, we want the row that has the minimum PWFO metric **m(ru-p)** in the in-sample section**. m(ru-p) = Median of all Trade's(Maximum Trade Run-up – Final Trade Profit).** This would produce a filter named **t50eqTrn |p<4|-m(ru-p).** This in-sample filter leaves only one row in the PWFO in-sample section with its associated strategy inputs and out-of-sample net profit in the out-of-sample section. This particular **t50eqTrn |p<4|-m(ru-p)** filter finds the strategy inputs parameters in each of the 262 in-sample sections and applies these inputs to each of the 262 out-of-sample sections. Using the filter in-sample strategy inputs on the 262 out-of-sample sections, the average out-of-sample performance is calculated. In addition many other important out-of-sample performance statistics for this filter are calculated and summarized. **Figure 3** shows such a filter computer run along with a small sample of other filter combinations that are constructed in a similar manner. **Row 4** of the sample output in **Figure 3** shows the results of the filter discussed above. A total of 3844 different metric filters were examined. We chose **Row 4** because it had a lower **BE**, **BLW, eqDD** and **Dev^2** and a higher **%Wtr** along with better statistics than the rows above it. More on this below and on how that number of filters combinations effect the probability that the filter chosen was or was not due to chance

Bootstrap Probability of Filter Results: Using modern "Bootstrap" techniques, we can calculate the probability of obtaining each filter's total out-of-sample ***net*** profits by chance. By ***net*** we mean subtracting the cost and slippage of all round trip trades from the total out-of-sample profits. Here is how the bootstrap technique is applied. Suppose as an example, we calculate the total out-of-sample net profits(tOnpNet) over all out-of-sample weeks for a given filter like above. A mirror filter is created. However, instead of picking an out-of-sample net profit(OSNP) from a row that the filter picks, the mirror filter picks a ***random*** row's OSNP in each of the 262 PWFO files. Suppose we repeat this random row section 5000 times. Each of the 5000 mirror filters will choose a random row's OSNP of their own in each of the 262 PWFO files. At the end, each of the 5000 mirror filters will have 262 ***random*** OSNP's picked from the rows of the 262 PWFO files. The sum of the 262 random OSNP picks for each of the 5000 mirror filters will generate a random total out-of-sample net profit(tOnpNet). The average and standard deviation of the 5000 mirror filter's different random tOnpNets will allow us to calculate the chance probability for each of ***our*** filter's tOnpNet. Thus given the mirror filter's bootstrap random tOnpNet average and standard deviation, we can calculate the probability of obtaining our filter's tOnpNet by pure chance alone. Since for this run we examined 3844(shown in Figure 3) different filters, we can calculate the expected number of cases that we could obtain by pure chance that would match or exceed the tOnpNet of the filter we have chosen or (3844) X (tOnpNet Probability). For our filter in row 4 in Figure 3 the expected number of cases that we could obtain by pure chance that would match or exceed the tOnpNet of $45890 of Row 4 is 3844 x 6.45 10^{-6} = 0.025. This is much less than one case so it is improbable that our result was due to pure chance.

Results

Table 1 on page 8 below presents a table of the 262 in-sample and out-of-sample windows, the selected optimum parameters and the weekly out-of-sample results using the filter described above.

Figure 1 presents a graph of the equity and net equity curves generated by using the filter on the 262 weeks ending 8/12/12 to 8/12/16. The equity curves are plotted from the Equity and Net Equity columns in Table 1. Plotted on the equity curves are 2nd Order Polynomial fits. 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. The grey line is the weekly TF prices superimposed on the equity chart.

Figure 2 Walk Forward Out-Of-Sample Performance for TF Fading Memory Polynomial Velocity Strategy
1 minute bar chart of TF from 6/24/16-6/28/2016

Figure 3 Partial output of the Walk Forward Metric Performance Explorer (WFME)
Run on the 262 PWFO files of the TF 1min bars Nth Order Fading Memory Velocity Strategy

Discussion of Strategy Performance

In Figure 3 Row 4 of the spreadsheet filter output are some statistics that are of interest for our filter. **BE** is the break even weeks. Assuming the trade average and standard deviation for this filter are from a normal distribution, this is how many weeks we need to trade this strategy so that we have a 98% probability that the equity paths after that number of weeks will be greater than zero. BE is 56.3 weeks for this filter. This means we would have to trade this strategy for at least 46.3weeks to have a 98% probability that our equity would be positive. Another interesting statistic is **Blw**. Blw is the maximum number of weeks the OSNP equity curve failed to make a new high. Blw is 29 weeks for this filter. This means that 29 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 ***pw, N, vup and 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 ***degree*** changes from a straight line velocity with ***degree=1*** to a 2nd, 3rd and 4th order velocity with ***degree= 2, 3 and 4***. The 3rd and 4th order velocities, due to the higher order components, change much faster than the straight line velocity. When the data gets very noisy with a lot of spurious price movements, it's better to have the velocity change slower filtering out the noisy data. During other times when the noise level is not as much it is better to have the velocity break its ***vup*** and ***vdn*** barriers faster to get onboard a trend faster. This is what the filter is doing. When there is a lot of noise in the in-sample section it switches to the 1st or 2nd order curve velocity. When the noise level is lower in the in-sample section, it switches to the faster changing 3rd or 4th order curve velocity.

Using this filter, the strategy was able to generate $45,890 net equity after commissions and slippage trading one TF contract for 262 weeks. Note $20 roundtrip commission and slippage was subtracted from each trade and no positions were carried overnight. The average net profit after commissions and slippage in the weeks that were traded was $259/week. The largest losing week was -$3100, the largest losing trade was $3340 and the largest drawdown was -$6340. The longest time between new equity highs was 29 weeks.

In observing Table 1 we can see that this strategy and filter made trades from a low of no trades/week to a high of 11 trades/week with an average of 2.6 trades/week on the weeks it did trade. The strategy seemed to wait for really strong trends and then initiate a buy or sell. There were many weeks that had no trades. Out of the 262 out-of-sample weeks the filter only traded 177 of those weeks or 67% of the time with 56% of all trades profitable. In observing the Equity Curve plot in Figure 1 we can see that the equity did quite well in both big up and down moves of the TF.

References

1. Efron, B., Tibshirani, R.J., (1993), "An Introduction to the Bootstrap", New York, Chapman & Hall/CRC.
2. Morrison, Norman "Introduction to Sequential Smoothing and Prediction", McGraw-Hill Book Company, New York, 1969.

Figure 1 Graph of Net Equity Curve Applying the Walk Forward Filter Each Week On TF 1min Bar Prices 08/12/11 – 8/12/16

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
The grey line is the TF Weekly Closing prices superimposed on the Equity Chart.

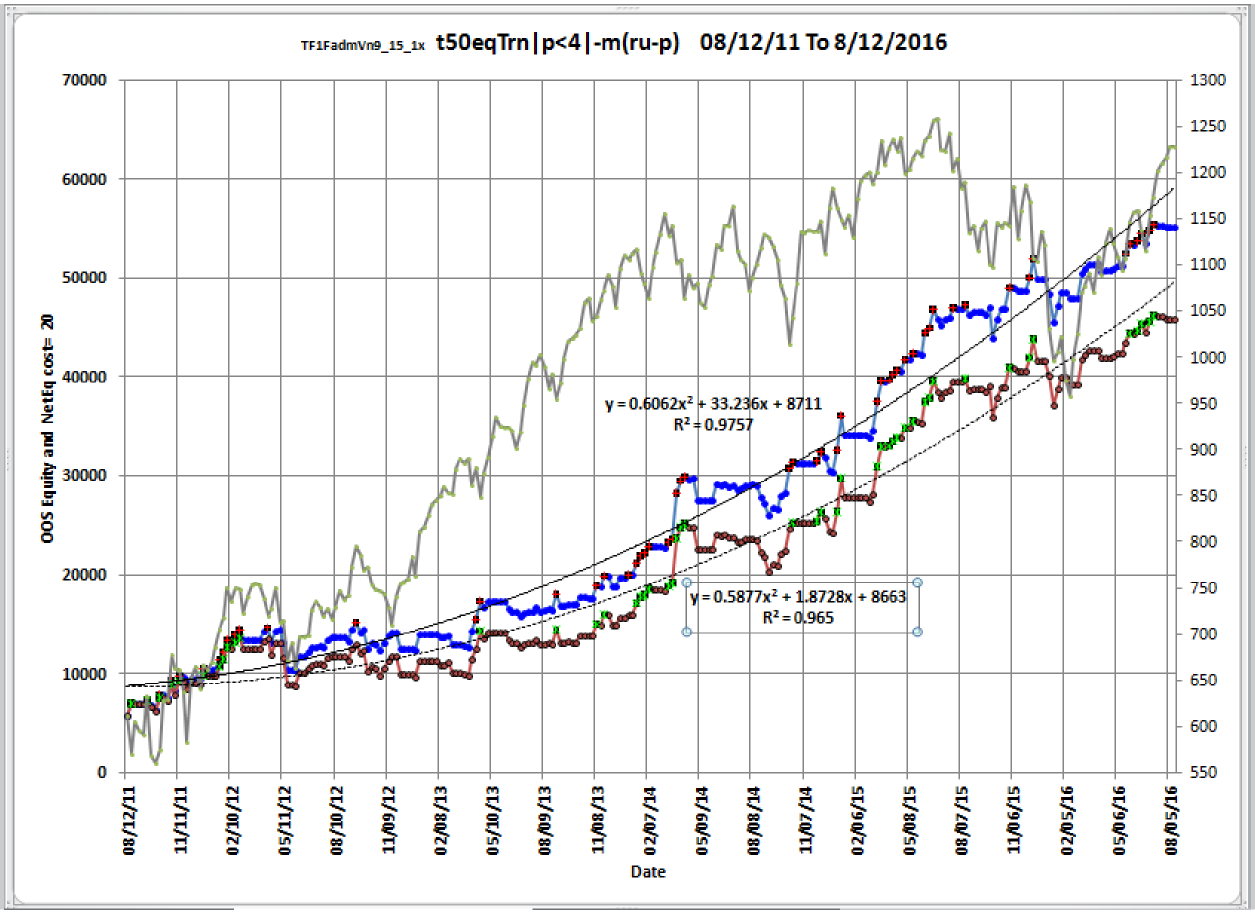


Figure 2 Walk Forward Out-Of-Sample Performance for TF Fading Memory Polynomial Velocity Strategy
1 minute bar chart of TF from 6/24/16-6/28/16

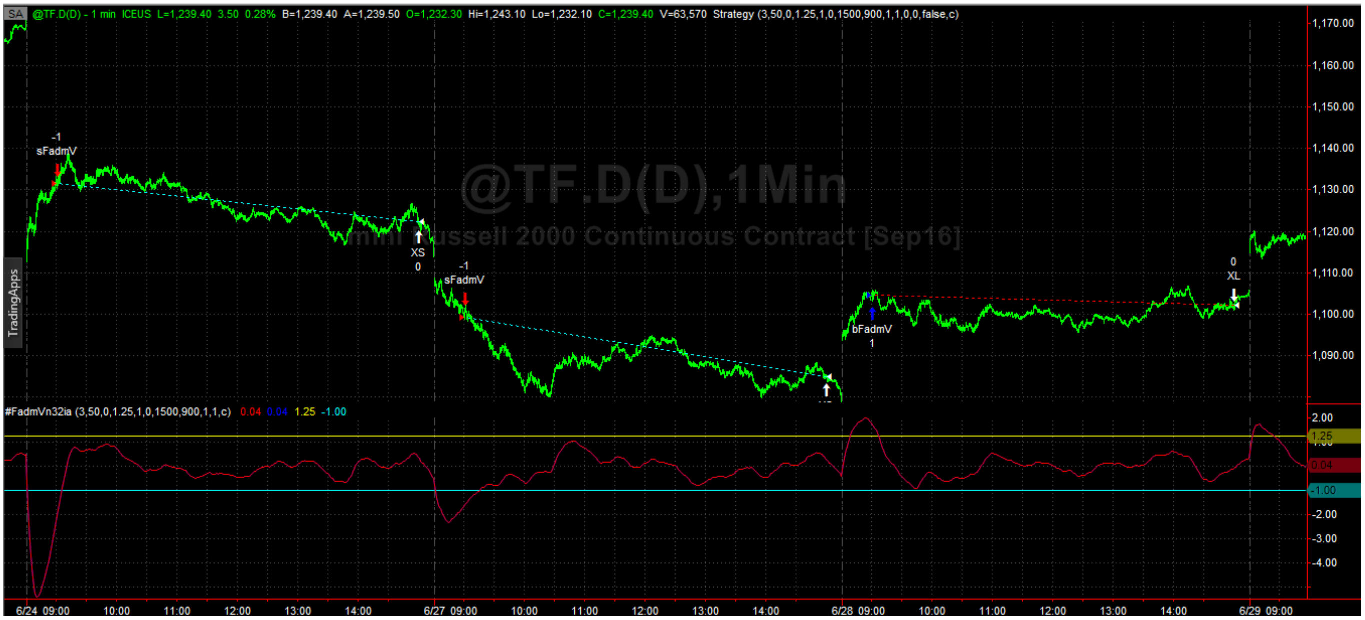
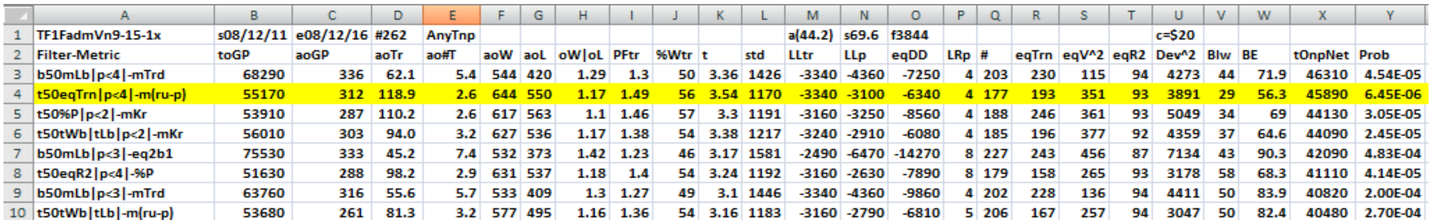


Figure 3_ Partial output of the Walk Forward Metric Performance Explorer (WFME)
TF1 min bars Nth Order Fading Memory Velocity Strategy



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
1	TF1FadmVn9-15-1x	s08/12/11	e08/12/16	#262	AnyTnp								a(44.2)	s69.6	f3844						c=$20				
2	Filter-Metric	toGP	aoGP	aoTr	ao#T	aoW	aoL	oW|oL	PFtr	%Wtr	t	std	LLtr	LLp	eqDD	LRp	#	eqTrn	eqV^2	eqR2	Dev^2	Blw	BE	tOnpNet	Prob
3	b50mLb|p<4|-mTrd	68290	336	62.1	5.4	544	420	1.29	1.3	50	3.36	1426	-3340	-4360	-7250	4	203	230	115	94	4273	44	71.9	46310	4.54E-05
4	t50eqTrn|p<4|-m(ru-p)	55170	312	118.9	2.6	644	550	1.17	1.49	56	3.54	1170	-3340	-3100	-6340	4	177	193	351	93	3891	29	56.3	45890	6.45E-06
5	t50%P|p<2|-mKr	53910	287	110.2	2.6	617	563	1.1	1.46	57	3.3	1191	-3160	-3250	-8560	4	188	246	361	93	5049	34	69	44130	3.05E-05
6	t50tWb|tLb|p<2|-mKr	56010	303	94.0	3.2	627	536	1.17	1.38	54	3.38	1217	-3240	-2910	-6080	4	185	196	377	92	4359	37	64.6	44090	2.45E-05
7	b50mLb|p<3|-eq2b1	75530	333	45.2	7.4	532	373	1.42	1.23	46	3.17	1581	-2490	-6470	-14270	8	227	243	456	87	7134	43	90.3	42090	4.83E-04
8	t50eqR2|p<4|-%P	51630	288	98.2	2.9	631	537	1.18	1.4	54	3.24	1192	-3160	-2630	-7890	8	179	158	265	93	3178	58	68.3	41110	4.14E-05
9	b50mLb|p<3|-mTrd	63760	316	55.6	5.7	533	409	1.3	1.27	49	3.1	1446	-3340	-4360	-9860	4	202	228	136	94	4411	50	83.9	40820	2.00E-04
10	t50tWb|tLb|-m(ru-p)	53680	261	81.3	3.2	577	495	1.16	1.36	54	3.16	1183	-3160	-2790	-6810	5	206	167	257	94	3047	50	82.4	40480	2.70E-04

The WFME Filter Output Columns are defined as follows:

Row 1 TF1FadmV is the strategy abbreviation, First OOS Week End Date(8/12/11), Last OOS Week End Date(8/12/16), **Number of weeks**(#262) **a**=weekly average of bootstrap random picks. **s**= weekly 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. Row10 filter **t50eqTrn|p<4|-m(ru-p)**

The t50eqTrn|p<4|-m(ru-p) filter produced the following average 262 week statistics on Row 4.

toGP = Total out-of-sample(oos) gross profit for these 262 weeks.

aoGP = Average oos gross profit for the 262 weeks

aoTr = Average oos profit per trade

ao#T = Average number of oos trades per week

aoW = Average of oos winning trades

aoL = Average of oos losing trades

oW|oL = aoW/aoL

PFtr = Sum of all oos winning trades/abs(sum of all oos losing trades)

%Wtr = The percentage of all oos trades that were profitable

t = The student t statistic for the 262 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 262 weekly oos profits

LLtr = The largest losing oos trade

LLp = The largest losing oos period(week)

eqDD = The oos equity drawdown

LRp = 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^2 = The ending velocity of 2nd 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.

tOnpNet = Total out-of-sample net profit(tOnpNet) minus the total trade cost.
tOnpNet=tOnp – (Number of trade weeks)*aOnT*Cost.

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

Table 1 Walk Forward Out-Of-Sample Performance Summary for TF Nth Order Fading Memory Polynomial Velocity Strategy

TF-1 min bars 8/12/2011 - 8/12/2016. The input values *degree(pw), N, vup, vdn* are the values found from applying the filter to the in-sample section optimization runs.

Filter= t50eqTrn |p<4|-m(ru-p) = PF<=4 then top 50 eqTrn and then minimum m(ru-p)

osnp = Weekly Out-of-sample gross profit in $
Equity = Running Sum of weekly out-of-sample gross profits $
NOnp$20 = Weekly Out-Of-Sample Net Profit in $ = **osnp-ont*20**.
NetEq = running sum of the weekly out-of-sample net profits in $
ollt = The largest losing trade in the out-of-sample section in $.
odd = The drawdown in the out-of-sample section in $.
ont = The number of trades in the out-of-sample week.
pw= degree, degree=1 for straight line velocity, degree=2 for 2nd order velocity, etc.
N = N the lookback period
vup, the threshold amount that velocity has to be greater than to issue a buy signal
vdn, the threshold amount that velocity has to be less than to issue a sell signal
Note: Blank rows indicate that no out-of-sample trades were made that week

In-Sample Dates			Out-Of-Sample Dates			osnp	Eq	NOnp$20	NetEq	ont	ollt	odd	pw	N	vup	vdn
07/07/11	to	08/05/11	08/08/11	to	08/12/11	5800	5800	5700	5700	5	-1230	-1230	1	20	2.25	0.5
07/14/11	to	08/12/11	08/15/11	to	08/19/11	1220	7020	1200	6900	1	0	0	4	70	1.75	2.25
07/21/11	to	08/19/11	08/22/11	to	08/26/11		7020		6900				4	70	1.75	2.5
07/28/11	to	08/26/11	08/29/11	to	09/02/11		7020		6900				2	20	2	3.25
08/04/11	to	09/02/11	09/05/11	to	09/09/11		7020		6900				2	20	2	3.25
08/11/11	to	09/09/11	09/12/11	to	09/16/11	390	7410	370	7270	1	0	0	1	50	1	2.75
08/18/11	to	09/16/11	09/19/11	to	09/23/11	(600)	6810	(620)	6650	1	-600	-600	1	40	0.75	3
08/25/11	to	09/23/11	09/26/11	to	09/30/11	(440)	6370	(480)	6170	2	-360	-440	4	70	1	2
09/01/11	to	09/30/11	10/03/11	to	10/07/11	1490	7860	1350	7520	7	-1520	-2260	2	70	1	0.75
09/08/11	to	10/07/11	10/10/11	to	10/14/11		7860		7520				2	20	1.5	1.75
09/15/11	to	10/14/11	10/17/11	to	10/21/11	(270)	7590	(290)	7230	1	-270	-270	2	20	1.5	1.75
09/22/11	to	10/21/11	10/24/11	to	10/28/11	1720	9310	1700	8930	1	0	0	4	70	1.5	1.5
09/29/11	to	10/28/11	10/31/11	to	11/04/11	(1030)	8280	(1050)	7880	1	-1030	-1030	4	50	1.25	1.5
10/06/11	to	11/04/11	11/07/11	to	11/11/11	1420	9700	1360	9240	3	0	0	2	60	0.75	2.25
10/13/11	to	11/11/11	11/14/11	to	11/18/11		9700		9240				2	40	1.75	1.25
10/20/11	to	11/18/11	11/21/11	to	11/25/11	(680)	9020	(720)	8520	2	-860	-860	2	70	0.5	1.25
10/27/11	to	11/25/11	11/28/11	to	12/02/11	550	9570	490	9010	3	-1090	-1090	2	50	0.75	2
11/03/11	to	12/02/11	12/05/11	to	12/09/11		9570		9010				4	50	1	1
11/10/11	to	12/09/11	12/12/11	to	12/16/11	(60)	9510	(120)	8890	3	-450	-450	1	70	1.25	0.75
11/17/11	to	12/16/11	12/19/11	to	12/23/11	950	10460	930	9820	1	0	0	2	40	1.25	1.25
11/24/11	to	12/23/11	12/26/11	to	12/30/11		10460		9820				3	60	2.75	0.75
12/01/11	to	12/30/11	01/02/12	to	01/06/12		10460		9820				1	20	1	1.25
12/08/11	to	01/06/12	01/09/12	to	01/13/12		10460		9820				3	70	2	1
12/15/11	to	01/13/12	01/16/12	to	01/20/12	970	11430	950	10770	1	0	0	2	30	0.5	0.75
12/22/11	to	01/20/12	01/23/12	to	01/27/12	660	12090	640	11410	1	0	0	4	60	0.5	0.75
12/29/11	to	01/27/12	01/30/12	to	02/03/12	1290	13380	1230	12640	3	0	0	3	50	0.5	1
01/05/12	to	02/03/12	02/06/12	to	02/10/12	(140)	13240	(180)	12460	2	-130	-140	4	70	0.5	0.75
01/12/12	to	02/10/12	02/13/12	to	02/17/12	710	13950	670	13130	2	0	0	1	60	0.5	1
01/19/12	to	02/17/12	02/20/12	to	02/24/12	470	14420	450	13580	1	0	0	2	70	0.5	1
01/26/12	to	02/24/12	02/27/12	to	03/02/12	(1060)	13360	(1140)	12440	4	-830	-1590	3	40	0.5	1
02/02/12	to	03/02/12	03/05/12	to	03/09/12		13360		12440				3	40	1	0.75
02/09/12	to	03/09/12	03/12/12	to	03/16/12		13360		12440				2	60	1.25	1
02/16/12	to	03/16/12	03/19/12	to	03/23/12		13360		12440				2	40	1	0.75

In-Sample Dates			Out-Of-Sample Dates			osnp	Eq	NOnp$20	NetEq	ont	ollt	odd	pw	N	vup	vdn
02/23/12	to	03/23/12	03/26/12	to	03/30/12		13360		12440				2	40	1	0.75
03/01/12	to	03/30/12	04/02/12	to	04/06/12	910	14270	850	13290	3	-320	-320	2	50	0.25	0.75
03/08/12	to	04/06/12	04/09/12	to	04/13/12	350	14620	270	13560	4	-400	-640	1	40	0.25	1
03/15/12	to	04/13/12	04/16/12	to	04/20/12	(1510)	13110	(1610)	11950	5	-1020	-1770	3	60	0.25	0.75
03/22/12	to	04/20/12	04/23/12	to	04/27/12	1250	14360	1130	13080	6	-260	-260	2	50	0.25	1.5
03/29/12	to	04/27/12	04/30/12	to	05/04/12	180	14540	100	13180	4	-700	-700	1	60	1.5	0.5
04/05/12	to	05/04/12	05/07/12	to	05/11/12	(1480)	13060	(1520)	11660	2	-980	-1480	1	60	1.5	0.5
04/12/12	to	05/11/12	05/14/12	to	05/18/12	(2630)	10430	(2690)	8970	3	-1650	-2630	3	30	0.5	1.5
04/19/12	to	05/18/12	05/21/12	to	05/25/12		10430		8970				3	70	1.25	0.75
04/26/12	to	05/25/12	05/28/12	to	06/01/12	(60)	10370	(160)	8810	5	-1280	-1630	1	50	0.5	0.75
05/03/12	to	06/01/12	06/04/12	to	06/08/12	1420	11790	1360	10170	3	0	0	3	70	0.75	1
05/10/12	to	06/08/12	06/11/12	to	06/15/12		11790		10170				1	60	0.5	1
05/17/12	to	06/15/12	06/18/12	to	06/22/12	390	12180	370	10540	1	0	0	1	60	0.5	1
05/24/12	to	06/22/12	06/25/12	to	06/29/12	420	12600	360	10900	3	-750	-750	2	40	0.75	0.75
05/31/12	to	06/29/12	07/02/12	to	07/06/12	110	12710	90	10990	1	0	0	2	70	1	0.75
06/07/12	to	07/06/12	07/09/12	to	07/13/12	30	12740	10	11000	1	0	0	2	70	0.5	1.25
06/14/12	to	07/13/12	07/16/12	to	07/20/12	(20)	12720	(120)	10880	5	-320	-670	3	40	0.25	1
06/21/12	to	07/20/12	07/23/12	to	07/27/12	770	13490	670	11550	5	-630	-1160	2	30	0.25	1
06/28/12	to	07/27/12	07/30/12	to	08/03/12	260	13750	240	11790	1	0	0	1	50	0.5	1
07/05/12	to	08/03/12	08/06/12	to	08/10/12		13750		11790				2	30	0.75	1
07/12/12	to	08/10/12	08/13/12	to	08/17/12		13750		11790				2	30	0.75	1
07/19/12	to	08/17/12	08/20/12	to	08/24/12		13750		11790				1	20	0.75	1
07/26/12	to	08/24/12	08/27/12	to	08/31/12	(490)	13260	(510)	11280	1	-490	-490	4	30	0.75	0.75
08/02/12	to	08/31/12	09/03/12	to	09/07/12	1260	14520	1160	12440	5	-430	-430	3	70	0.25	0.5
08/09/12	to	09/07/12	09/10/12	to	09/14/12	620	15140	580	13020	2	0	0	1	40	0.5	0.75
08/16/12	to	09/14/12	09/17/12	to	09/21/12	(910)	14230	(950)	12070	2	-510	-910	1	40	0.5	0.5
08/23/12	to	09/21/12	09/24/12	to	09/28/12	240	14470	220	12290	1	0	0	4	40	0.5	1
08/30/12	to	09/28/12	10/01/12	to	10/05/12	(1890)	12580	(1990)	10300	5	-1190	-1890	4	30	0.5	0.75
09/06/12	to	10/05/12	10/08/12	to	10/12/12	470	13050	450	10750	1	0	0	4	20	1	0.5
09/13/12	to	10/12/12	10/15/12	to	10/19/12	(110)	12940	(210)	10540	5	-660	-970	4	40	0.75	0.25
09/20/12	to	10/19/12	10/22/12	to	10/26/12	(650)	12290	(670)	9870	1	-650	-650	1	50	0.75	0.5
09/27/12	to	10/26/12	10/29/12	to	11/02/12	790	13080	750	10620	2	0	0	1	50	0.75	0.5
10/04/12	to	11/02/12	11/05/12	to	11/09/12	730	13810	690	11310	2	-70	-70	1	30	1	0.5
10/11/12	to	11/09/12	11/12/12	to	11/16/12	400	14210	380	11690	1	0	0	2	40	0.75	0.75
10/18/12	to	11/16/12	11/19/12	to	11/23/12		14210		11690				3	20	0.75	1
10/25/12	to	11/23/12	11/26/12	to	11/30/12	(1670)	12540	(1710)	9980	2	-1270	-1670	3	50	1.25	0.5
11/01/12	to	11/30/12	12/03/12	to	12/07/12	(10)	12530	(30)	9950	1	-10	-10	4	40	0.5	1
11/08/12	to	12/07/12	12/10/12	to	12/14/12		12530		9950				3	20	0.75	1
11/15/12	to	12/14/12	12/17/12	to	12/21/12		12530		9950				2	60	0.5	1
11/22/12	to	12/21/12	12/24/12	to	12/28/12	(150)	12380	(230)	9720	4	-540	-540	4	50	0.25	0.75
11/29/12	to	12/28/12	12/31/12	to	01/04/13	1660	14040	1600	11320	3	-10	-10	4	40	0.5	0.75
12/06/12	to	01/04/13	01/07/13	to	01/11/13		14040		11320				4	40	0.5	1
12/13/12	to	01/11/13	01/14/13	to	01/18/13		14040		11320				3	20	0.5	0.75
12/20/12	to	01/18/13	01/21/13	to	01/25/13		14040		11320				3	30	0.5	0.75
12/27/12	to	01/25/13	01/28/13	to	02/01/13		14040		11320				3	30	0.5	0.75
01/03/13	to	02/01/13	02/04/13	to	02/08/13	(370)	13670	(390)	10930	1	-370	-370	3	30	0.5	0.5
01/10/13	to	02/08/13	02/11/13	to	02/15/13		13670		10930				4	70	0.5	0.75
01/17/13	to	02/15/13	02/18/13	to	02/22/13	220	13890	180	11110	2	-30	-30	4	70	0.5	0.75
01/24/13	to	02/22/13	02/25/13	to	03/01/13	(920)	12970	(960)	10150	2	-1280	-1280	1	50	0.75	0.5
01/31/13	to	03/01/13	03/04/13	to	03/08/13		12970		10150				4	20	0.75	1
02/07/13	to	03/08/13	03/11/13	to	03/15/13		12970		10150				4	40	0.75	0.5
02/14/13	to	03/15/13	03/18/13	to	03/22/13	(190)	12780	(210)	9940	1	-190	-190	4	60	0.75	0.5
02/21/13	to	03/22/13	03/25/13	to	03/29/13	(170)	12610	(190)	9750	1	-170	-170	4	20	0.75	1
02/28/13	to	03/29/13	04/01/13	to	04/05/13	1780	14390	1740	11490	2	0	0	3	30	1	0.5

In-Sample Dates			Out-Of-Sample Dates			osnp	Eq	NOnp$20	NetEq	ont	ollt	odd	pw	N	vup	vdn
03/07/13	to	04/05/13	04/08/13	to	04/12/13	1100	15490	1060	12550	2	0	0	4	60	0.5	1
03/14/13	to	04/12/13	04/15/13	to	04/19/13	1820	17310	1760	14310	3	-410	-410	4	60	0.5	1
03/21/13	to	04/19/13	04/22/13	to	04/26/13	(630)	16680	(710)	13600	4	-1250	-1250	2	50	0.5	1
03/28/13	to	04/26/13	04/29/13	to	05/03/13	580	17260	560	14160	1	0	0	4	20	0.75	1.5
04/04/13	to	05/03/13	05/06/13	to	05/10/13		17260		14160				4	20	0.75	1.5
04/11/13	to	05/10/13	05/13/13	to	05/17/13		17260		14160				4	20	0.75	1.5
04/18/13	to	05/17/13	05/20/13	to	05/24/13		17260		14160				2	40	1.75	1
04/25/13	to	05/24/13	05/27/13	to	05/31/13		17260		14160				4	40	0.75	1
05/02/13	to	05/31/13	06/03/13	to	06/07/13	(730)	16530	(810)	13350	4	-770	-1470	4	40	0.75	0.75
05/09/13	to	06/07/13	06/10/13	to	06/14/13	(220)	16310	(240)	13110	1	-220	-220	4	40	0.75	1.5
05/16/13	to	06/14/13	06/17/13	to	06/21/13		16310		13110				4	40	0.75	1.5
05/23/13	to	06/21/13	06/24/13	to	06/28/13	(430)	15880	(450)	12660	1	-430	-430	2	60	2	1
05/30/13	to	06/28/13	07/01/13	to	07/05/13	280	16160	260	12920	1	0	0	2	50	0.75	1.5
06/06/13	to	07/05/13	07/08/13	to	07/12/13	150	16310	130	13050	1	0	0	1	60	0.75	1.25
06/13/13	to	07/12/13	07/15/13	to	07/19/13		16310		13050				3	40	0.5	1
06/20/13	to	07/19/13	07/22/13	to	07/26/13	440	16750	420	13470	1	0	0	4	30	0.5	1
06/27/13	to	07/26/13	07/29/13	to	08/02/13	(470)	16280	(510)	12960	2	-750	-750	4	30	0.5	0.75
07/04/13	to	08/02/13	08/05/13	to	08/09/13	80	16360	60	13020	1	0	0	4	50	0.5	0.75
07/11/13	to	08/09/13	08/12/13	to	08/16/13	240	16600	120	13140	6	-290	-500	4	60	0.5	0.25
07/18/13	to	08/16/13	08/19/13	to	08/23/13	(120)	16480	(200)	12940	4	-640	-640	4	30	0.75	0.5
07/25/13	to	08/23/13	08/26/13	to	08/30/13	1560	18040	1480	14420	4	-390	-390	1	40	0.25	0.75
08/01/13	to	08/30/13	09/02/13	to	09/06/13	(1130)	16910	(1230)	13190	5	-860	-1130	1	60	0.25	0.5
08/08/13	to	09/06/13	09/09/13	to	09/13/13		16910		13190				4	20	1.25	1.25
08/15/13	to	09/13/13	09/16/13	to	09/20/13	50	16960	30	13220	1	0	0	2	50	1.25	0.5
08/22/13	to	09/20/13	09/23/13	to	09/27/13	20	16980	(40)	13180	3	-250	-250	2	40	1.75	0.5
08/29/13	to	09/27/13	09/30/13	to	10/04/13		16980		13180				4	70	1	1.25
09/05/13	to	10/04/13	10/07/13	to	10/11/13	740	17720	720	13900	1	0	0	3	60	1	1.25
09/12/13	to	10/11/13	10/14/13	to	10/18/13		17720		13900				3	50	1.75	0.75
09/19/13	to	10/18/13	10/21/13	to	10/25/13	(40)	17680	(60)	13840	1	-40	-40	3	50	1	0.75
09/26/13	to	10/25/13	10/28/13	to	11/01/13		17680		13840				1	20	0.75	1
10/03/13	to	11/01/13	11/04/13	to	11/08/13	1190	18870	1150	14990	2	0	0	1	70	0.75	0.75
10/10/13	to	11/08/13	11/11/13	to	11/15/13		18870		14990				2	20	1.25	1
10/17/13	to	11/15/13	11/18/13	to	11/22/13	1000	19870	980	15970	1	0	0	3	60	0.75	1
10/24/13	to	11/22/13	11/25/13	to	11/29/13		19870		15970				3	60	0.75	0.75
10/31/13	to	11/29/13	12/02/13	to	12/06/13	(980)	18890	(1020)	14950	2	-820	-980	3	60	0.75	0.75
11/07/13	to	12/06/13	12/09/13	to	12/13/13		18890		14950				3	70	0.75	1
11/14/13	to	12/13/13	12/16/13	to	12/20/13	790	19680	750	15700	2	0	0	1	70	0.75	0.75
11/21/13	to	12/20/13	12/23/13	to	12/27/13		19680		15700				4	70	1.25	0.75
11/28/13	to	12/27/13	12/30/13	to	01/03/14	290	19970	270	15970	1	0	0	4	70	1.25	0.75
12/05/13	to	01/03/14	01/06/14	to	01/10/14		19970		15970				2	20	2.25	0.75
12/12/13	to	01/10/14	01/13/14	to	01/17/14	1160	21130	1120	17090	2	0	0	1	70	0.25	1
12/19/13	to	01/17/14	01/20/14	to	01/24/14	710	21840	650	17740	3	-130	-130	1	70	0.25	1
12/26/13	to	01/24/14	01/27/14	to	01/31/14	320	22160	220	17960	5	-610	-610	3	70	0.5	1.5
01/02/14	to	01/31/14	02/03/14	to	02/07/14	680	22840	620	18580	3	-60	-60	4	50	0.75	1.25
01/09/14	to	02/07/14	02/10/14	to	02/14/14		22840		18580				3	40	1	0.75
01/16/14	to	02/14/14	02/17/14	to	02/21/14		22840		18580				4	50	1	0.75
01/23/14	to	02/21/14	02/24/14	to	02/28/14		22840		18580				1	60	1	1
01/30/14	to	02/28/14	03/03/14	to	03/07/14	(150)	22690	(190)	18390	2	-320	-320	1	70	0.75	0.5
02/06/14	to	03/07/14	03/10/14	to	03/14/14	500	23190	440	18830	3	-260	-380	3	20	0.75	1
02/13/14	to	03/14/14	03/17/14	to	03/21/14	380	23570	320	19150	3	-250	-250	3	20	0.75	0.75
02/20/14	to	03/21/14	03/24/14	to	03/28/14	4690	28260	4530	23680	8	-220	-220	4	30	0.75	0.5
02/27/14	to	03/28/14	03/31/14	to	04/04/14	1220	29480	1100	24780	6	-1070	-2200	4	20	1	0.5
03/06/14	to	04/04/14	04/07/14	to	04/11/14	450	29930	410	25190	2	-910	-910	3	20	1.75	1
03/13/14	to	04/11/14	04/14/14	to	04/18/14	(270)	29660	(290)	24900	1	-270	-270	3	20	1.75	1.25

In-Sample Dates			Out-Of-Sample Dates			osnp	Eq	NOnp$20	NetEq	ont	ollt	odd	pw	N	vup	vdn
03/20/14	to	04/18/14	04/21/14	to	04/25/14	60	29720	(20)	24880	4	-1040	-1040	1	60	1.25	0.25
03/27/14	to	04/25/14	04/28/14	to	05/02/14	(2220)	27500	(2300)	22580	4	-680	-2220	1	70	0.75	1
04/03/14	to	05/02/14	05/05/14	to	05/09/14		27500		22580				1	30	1	2
04/10/14	to	05/09/14	05/12/14	to	05/16/14		27500		22580				4	30	1.25	1.75
04/17/14	to	05/16/14	05/19/14	to	05/23/14		27500		22580				4	30	1.25	1.75
04/24/14	to	05/23/14	05/26/14	to	05/30/14		27500		22580				4	30	1.25	1.75
05/01/14	to	05/30/14	06/02/14	to	06/06/14	1630	29130	1490	24070	7	-760	-850	1	40	0.25	0.5
05/08/14	to	06/06/14	06/09/14	to	06/13/14	(160)	28970	(180)	23890	1	-160	-160	4	60	0.75	1.5
05/15/14	to	06/13/14	06/16/14	to	06/20/14	140	29110	120	24010	1	0	0	1	30	0.75	1.25
05/22/14	to	06/20/14	06/23/14	to	06/27/14	(180)	28930	(240)	23770	3	-1540	-1540	4	50	0.5	1
05/29/14	to	06/27/14	06/30/14	to	07/04/14	80	29010	60	23830	1	0	0	1	50	1	0.75
06/05/14	to	07/04/14	07/07/14	to	07/11/14	(390)	28620	(530)	23300	7	-610	-820	2	40	0.25	0.75
06/12/14	to	07/11/14	07/14/14	to	07/18/14	110	28730	10	23310	5	-360	-1020	1	50	0.25	0.75
06/19/14	to	07/18/14	07/21/14	to	07/25/14	310	29040	290	23600	1	0	0	4	30	1.5	0.75
06/26/14	to	07/25/14	07/28/14	to	08/01/14		29040		23600				3	30	0.75	1.25
07/03/14	to	08/01/14	08/04/14	to	08/08/14	90	29130	50	23650	2	-400	-400	3	20	1	0.75
07/10/14	to	08/08/14	08/11/14	to	08/15/14	(110)	29020	(130)	23520	1	-110	-110	3	60	1	1
07/17/14	to	08/15/14	08/18/14	to	08/22/14	(1230)	27790	(1250)	22270	1	-1230	-1230	3	20	1	0.75
07/24/14	to	08/22/14	08/25/14	to	08/29/14	(500)	27290	(520)	21750	1	-500	-500	2	50	0.75	1.25
07/31/14	to	08/29/14	09/01/14	to	09/05/14	(1330)	25960	(1410)	20340	4	-1090	-1520	3	30	0.25	1.25
08/07/14	to	09/05/14	09/08/14	to	09/12/14	780	26740	720	21060	3	0	0	4	40	0.5	1.25
08/14/14	to	09/12/14	09/15/14	to	09/19/14	(80)	26660	(160)	20900	4	-310	-390	3	20	0.5	1.5
08/21/14	to	09/19/14	09/22/14	to	09/26/14	1310	27970	1270	22170	2	0	0	4	70	1	0.75
08/28/14	to	09/26/14	09/29/14	to	10/03/14	310	28280	210	22380	5	-880	-930	3	40	1.25	0.5
09/04/14	to	10/03/14	10/06/14	to	10/10/14	2480	30760	2360	24740	6	-2350	-2350	3	40	1.25	0.5
09/11/14	to	10/10/14	10/13/14	to	10/17/14	550	31310	510	25250	2	-700	-700	4	40	1	1.5
09/18/14	to	10/17/14	10/20/14	to	10/24/14		31310		25250				4	30	1.25	1.75
09/25/14	to	10/24/14	10/27/14	to	10/31/14		31310		25250				4	30	1.25	1.75
10/02/14	to	10/31/14	11/03/14	to	11/07/14		31310		25250				4	30	1.25	1.75
10/09/14	to	11/07/14	11/10/14	to	11/14/14		31310		25250				4	30	1.25	1.75
10/16/14	to	11/14/14	11/17/14	to	11/21/14		31310		25250				4	20	0.75	1.75
10/23/14	to	11/21/14	11/24/14	to	11/28/14	170	31480	150	25400	1	0	0	4	50	0.75	0.75
10/30/14	to	11/28/14	12/01/14	to	12/05/14	910	32390	890	26290	1	0	0	2	40	1	0.75
11/06/14	to	12/05/14	12/08/14	to	12/12/14	(490)	31900	(530)	25760	2	-950	-950	1	50	1	1
11/13/14	to	12/12/14	12/15/14	to	12/19/14	(1320)	30580	(1360)	24400	2	-1840	-1840	1	70	1	1
11/20/14	to	12/19/14	12/22/14	to	12/26/14	(190)	30390	(210)	24190	1	-190	-190	2	30	1.25	0.5
11/27/14	to	12/26/14	12/29/14	to	01/02/15	2190	32580	2150	26340	2	0	0	4	20	2	0.75
12/04/14	to	01/02/15	01/05/15	to	01/09/15	3500	36080	3420	29760	4	0	0	1	70	0.75	0.25
12/11/14	to	01/09/15	01/12/15	to	01/16/15	(1940)	34140	(1960)	27800	1	-1940	-1940	1	40	1.25	1.5
12/18/14	to	01/16/15	01/19/15	to	01/23/15		34140		27800				3	50	1.75	1.25
12/25/14	to	01/23/15	01/26/15	to	01/30/15		34140		27800				4	20	1.25	1.75
01/01/15	to	01/30/15	02/02/15	to	02/06/15		34140		27800				4	20	1.25	1.75
01/08/15	to	02/06/15	02/09/15	to	02/13/15		34140		27800				4	30	1	1.5
01/15/15	to	02/13/15	02/16/15	to	02/20/15		34140		27800				3	20	1	1.5
01/22/15	to	02/20/15	02/23/15	to	02/27/15	(330)	33810	(350)	27450	1	-330	-330	4	30	0.75	1.5
01/29/15	to	02/27/15	03/02/15	to	03/06/15	710	34520	630	28080	4	0	0	4	30	0.5	1.5
02/05/15	to	03/06/15	03/09/15	to	03/13/15	2960	37480	2860	30940	5	0	0	2	20	0.25	1.25
02/12/15	to	03/13/15	03/16/15	to	03/20/15	2110	39590	2010	32950	5	-40	-40	3	20	0.25	1
02/19/15	to	03/20/15	03/23/15	to	03/27/15		39590		32950				2	70	0.75	1.75
02/26/15	to	03/27/15	03/30/15	to	04/03/15	90	39680	50	33000	2	-390	-390	2	70	0.75	1.75
03/05/15	to	04/03/15	04/06/15	to	04/10/15	530	40210	470	33470	3	0	0	3	60	0.5	1.5
03/12/15	to	04/10/15	04/13/15	to	04/17/15	400	40610	360	33830	2	-110	-110	2	70	0.5	1.25
03/19/15	to	04/17/15	04/20/15	to	04/24/15		40610		33830				4	70	1	1
03/26/15	to	04/24/15	04/27/15	to	05/01/15	1110	41720	1030	34860	4	-1560	-1560	4	70	1	1

In-Sample Dates			Out-Of-Sample Dates			osnp	Eq	NOnp$20	NetEq	ont	ollt	odd	pw	N	vup	vdn
04/02/15	to	05/01/15	05/04/15	to	05/08/15		41720		34860				2	30	1	1.25
04/09/15	to	05/08/15	05/11/15	to	05/15/15	660	42380	640	35500	1	0	0	2	30	1	1.25
04/16/15	to	05/15/15	05/18/15	to	05/22/15		42380		35500				4	40	1	1.25
04/23/15	to	05/22/15	05/25/15	to	05/29/15	(140)	42240	(160)	35340	1	-140	-140	2	20	0.75	1.75
04/30/15	to	05/29/15	06/01/15	to	06/05/15	2240	44480	2140	37480	5	-860	-860	4	30	0.5	1.25
05/07/15	to	06/05/15	06/08/15	to	06/12/15	460	44940	360	37840	5	-440	-440	3	30	0.25	1.25
05/14/15	to	06/12/15	06/15/15	to	06/19/15	1820	46760	1720	39560	5	-510	-510	3	40	0.25	1.5
05/21/15	to	06/19/15	06/22/15	to	06/26/15	(970)	45790	(1030)	38530	3	-780	-1020	2	30	0.25	1.5
05/28/15	to	06/26/15	06/29/15	to	07/03/15	(600)	45190	(620)	37910	1	-600	-600	2	70	0.75	1.75
06/04/15	to	07/03/15	07/06/15	to	07/10/15	650	45840	570	38480	4	-760	-760	2	60	0.5	1.75
06/11/15	to	07/10/15	07/13/15	to	07/17/15	200	46040	180	38660	1	0	0	1	50	1	1.5
06/18/15	to	07/17/15	07/20/15	to	07/24/15	920	46960	880	39540	2	0	0	4	60	1.25	1
06/25/15	to	07/24/15	07/27/15	to	07/31/15		46960		39540				2	20	1	1
07/02/15	to	07/31/15	08/03/15	to	08/07/15		46960		39540				2	20	1	1
07/09/15	to	08/07/15	08/10/15	to	08/14/15	270	47230	250	39790	1	0	0	4	20	1	1
07/16/15	to	08/14/15	08/17/15	to	08/21/15	(1000)	46230	(1140)	38650	7	-740	-2150	4	20	1	1
07/23/15	to	08/21/15	08/24/15	to	08/28/15	340	46570	120	38770	11	-1990	-3780	4	20	1.5	1
07/30/15	to	08/28/15	08/31/15	to	09/04/15		46570		38770				4	60	1.75	1
08/06/15	to	09/04/15	09/07/15	to	09/11/15		46570		38770				3	50	1.75	1
08/13/15	to	09/11/15	09/14/15	to	09/18/15	(270)	46300	(290)	38480	1	-270	-270	3	60	1.5	0.75
08/20/15	to	09/18/15	09/21/15	to	09/25/15	710	47010	610	39090	5	-820	-820	3	60	1.5	0.75
08/27/15	to	09/25/15	09/28/15	to	10/02/15	(3100)	43910	(3160)	35930	3	-3160	-4720	2	70	1.25	1
09/03/15	to	10/02/15	10/05/15	to	10/09/15	1930	45840	1890	37820	2	0	0	1	70	0.75	1
09/10/15	to	10/09/15	10/12/15	to	10/16/15	1100	46940	1080	38900	1	0	0	1	50	0.75	1.5
09/17/15	to	10/16/15	10/19/15	to	10/23/15		46940		38900				4	60	1	1.5
09/24/15	to	10/23/15	10/26/15	to	10/30/15	2050	48990	2030	40930	1	0	0	1	70	0.5	1
10/01/15	to	10/30/15	11/02/15	to	11/06/15		48990		40930				1	60	0.75	1
10/08/15	to	11/06/15	11/09/15	to	11/13/15	(360)	48630	(380)	40550	1	-360	-360	3	30	0.75	2.25
10/15/15	to	11/13/15	11/16/15	to	11/20/15		48630		40550				4	20	1	2
10/22/15	to	11/20/15	11/23/15	to	11/27/15		48630		40550				4	20	1	2
10/29/15	to	11/27/15	11/30/15	to	12/04/15	1420	50050	1400	41950	1	0	0	1	70	1	0.5
11/05/15	to	12/04/15	12/07/15	to	12/11/15	1890	51940	1810	43760	4	-410	-410	4	70	1	0.75
11/12/15	to	12/11/15	12/14/15	to	12/18/15	(1970)	49970	(2070)	41690	5	-1290	-2660	3	70	1.25	0.5
11/19/15	to	12/18/15	12/21/15	to	12/25/15		49970		41690				1	60	0.75	1.5
11/26/15	to	12/25/15	12/28/15	to	01/01/16		49970		41690				4	30	0.75	1.25
12/03/15	to	01/01/16	01/04/16	to	01/08/16	(1510)	48460	(1590)	40100	4	-1770	-2310	3	60	0.75	1.25
12/10/15	to	01/08/16	01/11/16	to	01/15/16	(2860)	45600	(2960)	37140	5	-3340	-4460	1	50	0.5	2.5
12/17/15	to	01/15/16	01/18/16	to	01/22/16	1640	47240	1580	38720	3	-1110	-1190	2	30	1.25	1.75
12/24/15	to	01/22/16	01/25/16	to	01/29/16	1250	48490	1210	39930	2	0	0	4	20	1	1.75
12/31/15	to	01/29/16	02/01/16	to	02/05/16		48490		39930				4	70	1.75	1.75
01/07/16	to	02/05/16	02/08/16	to	02/12/16	(610)	47880	(630)	39300	1	-610	-610	4	70	1.75	1.75
01/14/16	to	02/12/16	02/15/16	to	02/19/16		47880		39300				4	70	1.75	2
01/21/16	to	02/19/16	02/22/16	to	02/26/16		47880		39300				1	70	1.25	2
01/28/16	to	02/26/16	02/29/16	to	03/04/16	2660	50540	2560	41860	5	-870	-870	1	60	0.25	2
02/04/16	to	03/04/16	03/07/16	to	03/11/16	440	50980	420	42280	1	0	0	4	30	0.75	1.25
02/11/16	to	03/11/16	03/14/16	to	03/18/16	450	51430	430	42710	1	0	0	2	40	1.25	1
02/18/16	to	03/18/16	03/21/16	to	03/25/16		51430		42710				2	40	1.25	1
02/25/16	to	03/25/16	03/28/16	to	04/01/16		51430		42710				4	50	0.75	1
03/03/16	to	04/01/16	04/04/16	to	04/08/16	(750)	50680	(790)	41920	2	-820	-820	4	50	0.75	1
03/10/16	to	04/08/16	04/11/16	to	04/15/16	90	50770	30	41950	3	-1330	-1330	1	60	0.75	1
03/17/16	to	04/15/16	04/18/16	to	04/22/16		50770		41950				3	70	1.25	1
03/24/16	to	04/22/16	04/25/16	to	04/29/16	230	51000	190	42140	2	0	0	1	20	1.25	0.75
03/31/16	to	04/29/16	05/02/16	to	05/06/16	290	51290	270	42410	1	0	0	3	70	1.25	1
04/07/16	to	05/06/16	05/09/16	to	05/13/16	10	51300	(30)	42380	2	-20	-20	3	70	0.75	1

In-Sample Dates			Out-Of-Sample Dates			osnp	Eq	NOnp$20	NetEq	ont	ollt	odd	pw	N	vup	vdn
04/14/16	to	05/13/16	05/16/16	to	05/20/16	1150	52450	1090	43470	3	-220	-220	1	70	0.5	1.25
04/21/16	to	05/20/16	05/23/16	to	05/27/16	960	53410	920	44390	2	-120	-120	2	70	0.75	0.5
04/28/16	to	05/27/16	05/30/16	to	06/03/16		53410		44390				1	20	0.75	1.5
05/05/16	to	06/03/16	06/06/16	to	06/10/16	260	53670	240	44630	1	0	0	1	70	1	1
05/12/16	to	06/10/16	06/13/16	to	06/17/16	810	54480	750	45380	3	-260	-260	4	40	0.5	1.5
05/19/16	to	06/17/16	06/20/16	to	06/24/16	(940)	53540	(960)	44420	1	-940	-940	4	30	0.75	1
05/26/16	to	06/24/16	06/27/16	to	07/01/16	1190	54730	1150	45570	2	-240	-240	3	50	1.25	1
06/02/16	to	07/01/16	07/04/16	to	07/08/16	650	55380	590	46160	3	-690	-690	1	30	0.75	3
06/09/16	to	07/08/16	07/11/16	to	07/15/16		55380		46160				1	70	1.25	1.5
06/16/16	to	07/15/16	07/18/16	to	07/22/16		55380		46160				1	70	1.5	1.25
06/23/16	to	07/22/16	07/25/16	to	07/29/16	(290)	55090	(310)	45850	1	-290	-290	2	30	0.75	1.25
06/30/16	to	07/29/16	08/01/16	to	08/05/16	80	55170	40	45890	2	-300	-300	4	40	0.75	1.25
07/07/16	to	08/05/16	08/08/16	to	08/12/16		55170		45890				1	70	0.75	1

Appendix 1: *n*th Order Fading Memory Polynomial Next Bar's Forecast Math

What is The Nth Order Fading Memory Polynomial ?

This is a mathematical technique that fits a nth order polynomial to the last ***T*** price bars but calculates the coefficients of the polynomial such that the error between the current nth order polynomial and the current bar is weighted much higher than the error between the price T bars ago and the value of the nth order polynomial T bars ago. As an example, if the latest price is at time t and the price made a turn at time bar t-10 , then we do not want prices prior to t-10 affecting the current polynomial fit as much. As will be shown the most familiar case of this fading memory technique is the exponential moving average. The fading memory technique is in contrast to the Least Squares Polynomial fit, which weights all past errors between the polynomial and the price bar equally.

Consider a time series x(t) where t is an integer value (a price bar number) like the number of days or minutes, etc from some starting time. Suppose we want to find at some given time some *n*th-degree polynomial that fits the data well at current and recent prices but ignores the fit as we move into the distant past. One way to construct this type of fit would be to weight the past data with a number that got smaller and smaller the further back in time we went. If we let the polynomial function be represented by the symbol **p(t-τ)** where **p(t-0)** is the current value of the polynomial, **p(t-1)** is the previous value of the polynomial, etc., then an error function can be formed that consists of the weighted sum of the squared difference between the price series **x(t-τ)** and the polynomial p(t-τ) given by

$$\textbf{error} = \sum \beta^{\tau}(x(t-\tau) - p(t-\tau))^2 \quad \tau=0 \text{ to } \infty \qquad (1)$$

where $0 < \beta < 1$ **and** β^{τ} is much much less than **1** for large **τ.**

It turns out that if we let the nth degree polynomial **p(t-τ)** be constructed as a linear combination of orthogonal polynomials called Meixner polynomials then minimizing the error with respect to the coefficients of the orthogonal polynomials yields the best estimate of **x(t-τ)** as **x$_{est}$(t-τ)** and given by the equation

$$x_{est}(t-\tau) = (1-\beta)\sum_{k=0}^{n} \beta^k b_{k,t} \Phi_k(t)|_{\tau} \qquad (2)$$

Where

$$\Phi_n(t) = \sum_{k=0}^{n} \binom{n}{k}\binom{t}{k} z^k$$

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

$$b_{j,t} = \sum_{k=0}^{\infty} \beta^k \Phi_j(k) x(t-k)$$

$$\mathbf{z = 1 - 1/\beta}$$

where n is the polynomial degree, $\Phi_k(\tau)$ are the Meixner polynomials of degree k (k=0 to n), and $\mathbf{b_k(t)}$ are the coefficients that minimize the error of equation (1). Generally the summation for $\mathbf{b_j(t)}$ can be terminated when β^k <<1.

Appendix 1: *n*th Order Fading Memory Polynomial Next Bar's Forecast Math

For the exact mathematical solutions that produce equation (2) and the mathematical descriptions of the Meixner polynomials refer to Reference 1.

To yield the 1 day ahead prediction the above equation becomes;

$$x_{est}(t+1)=(1-\beta)\sum \beta^k b_{k,t}\Phi_k(-1) \quad k=0 \text{ to } n \qquad (3)$$

After some algebraic manipulation with the Meixner polynomials the $b_{k,t}$ coefficients satisfy the following recursive relationship.(see Reference 1)

$$b_{k,t}=\beta\ b_{k,t-1}+b_{k-1,t}-b_{k-1,t-1}$$

One case is of immediate interest where the polynomial is a constant, that is *n*=0.

For this case the solution to equation (3) can be found after some algebraic manipulation to be:

$$X0_{est}=\beta*X0_{est}[1]+(1-\beta)*x(t) \qquad (4)$$

Where **X0$_{est}$[1]** is the previous estimated value, **x(t)** is the current bar's price and where the 0 in **X0$_{est}$** indicates that we are estimating a polynomial of degree 0 or simply a constant. If a change of variables is made letting **α=(1-β)** then equation (4) becomes:

$$X0_{est}=(1-\alpha)*X0_{est}[1]+\alpha*x(t) \qquad (5)$$

This is the familiar formula for the exponential moving average.

Higher orders of ***n*** don't yield such compact solutions as the case where ***n***=0 .equations

$$P_F(T+1) = (1-\beta)*[b_{0,t}\,\phi_{0|t=-1} + \beta b_{1,t}*\phi_{1|t=-1} + \beta^2 b_{2,t}*\phi_{2|t=-1} +\ldots+ \beta^n b_{n,t}*\phi_{n|t=-1}]$$

$$\text{Velocity} = (dP_F/dt)_{(T=-1)} = (1-\beta)[\beta b_{1,t}*(d\phi_1/dt)_{|t=-1} + \beta^2 b_{2,t}*(d\phi_2/dt)_{|t=-1} +\ldots+ \beta^n b_{n,t}*(d\phi_n/dt)_{|t=-1)}]$$

The nth Order Fading Memory Forecast Next Bar's Velocity Strategy Defined

The least squares forecast is constructed by solving for the coefficients **b_0, b_1, b_2, , , b_n** recursively at each bar using the last **T** bars of closing prices and the Discrete Orthogonal Meixner Polynomial equations above. Then **Velocity** = **dP$_F$(T+1)/dt** is constructed from the velocity equation above and plotted under the price chart. In general what we will be doing is following the plotted curve of **Velocity** which is calculated at each bar from the previous T bars. When the velocity is greater than a threshold amount ***vup*** we will go long. When the velocity is less than a 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.

Appendix 1: n^{th} Order Fading Memory Polynomial Next Bar's Forecast Math

References

1. Morrison, Norman “Introduction to Sequential Smoothing and Prediction", McGraw-Hill Book Company, New York, 1969.

Appendix 2 -The *Normalization Multiplier*

What is the Multiplier ?

The nth Order Fading Memory Polynomial, also called here an nth Order Polynomial, is the least square fit of a polynomial of the form $\mathbf{P_F(T+1) = (1-\beta)*[b_{0,t}\,\phi_{0|t=-1} + \beta b_{1,t}*\phi_{1|t=-1} + \beta^2 b_{2,t}*\phi_{2|t=-1} +\ldots+ \beta^n b_{n,t}*\phi_{n|t=-1}]}$ to a ***Fading*** number of past data points. Where **t** is discrete time bars. Time could be daily bars or one minute bars. We use the term "Fading Memory" because we weight the past data with a number that gets smaller and smaller the further back in time we go. It is assumed that the time bars occur at fixed intervals of time so tic bars would not be appropriate for this analysis. Suppose we want to find at some given time some *n*th-degree polynomial that fits the data well at current and recent prices but ignores the fit as we move into the distant past. One way to construct this type of fit would be to weight the past data with a number that got smaller and smaller the further back in time we went. This type of error minimization is mathematically solvable and is widely used in science and mathematics. Once the $\mathbf{b_n}$ coefficients are found then the next bar's estimate **(T+1)** of the nth order polynomial Acceleration and acceleration can be easily found by the equations below.

$$\textbf{Velocity} = (dP_F/dt)_{(T=-1)} = (1-\beta)[\beta b_{1,t} *(d\phi_1/dt)_{|t=-1} + \beta^2 b_{2,t} *(d\phi_2/dt)_{|t=-1} +\ldots+ \beta^n b_{n,t} *(d\phi_n/dt)_{|t=-1)}]$$

$$0<\beta<1$$

Please see the ***n*th Order Fading Memory Polynomial Next Bar's Forecast Math** section for a more detailed explanation.

For any tradable, the inputs to the polynomial are the **polynomial degree(aka Order)** and exponential decay constant β. When we plot the velocity we notice that the amplitude, and the maximum and minimum values of and standard deviation of the velocity vary quite significantly with different polynomial degree and β inputs.

Below is a table of the standard deviation of the 304156 calculated Velocity values for different **degree** and β inputs. The lookback period **N (number of bars of past data)** is approximately related to α **(α=1-β)** by the formula **α=2/(1+N)**. I use **N** because it gives a better understanding of how many bars of past data are approximately being used. We used 1 min bars of the E-Mini from 5/1/2012 to 5/1/2015 to generate this table.

ES 1 min bars Date Range 1120501 to 1150501
Total Number of Bars=304156 Norm=0
Trading Times Constraint Start Time=0 EndTime=0
FadmVn Multiplier to Scale Velocity pw and N Range to One Std

degree	N	Std	1/Std
1	20	0.095791	10.2624
1	30	0.078206	12.7867
1	40	0.0678338	14.7419
1	50	0.0607871	16.4509
1	60	0.0555952	17.9872
1	70	0.051562	19.3941
2	20	0.166382	6.01027
2	30	0.135156	7.39888
2	40	0.116914	8.55332
2	50	0.104594	9.56079
2	60	0.0955577	10.4649
2	70	0.0885688	11.2907
3	20	0.229464	4.35799
3	30	0.185563	5.38902
3	40	0.160108	6.2458
3	50	0.143001	6.99297
3	60	0.13048	7.66398
3	70	0.120815	8.27714

4	20	0.288656	3.46433
4	30	0.232499	4.30109
4	40	0.200125	4.99688
4	50	0.178495	5.60241
4	60	0.162712	6.14583
4	70	0.150529	6.64324

The standard deviation varies from 0.06 to 0.28. The problem may get worse when we want to find good inputs for other tradables. Other tradables, because of their scales and tick size have much different Velocity ranges then the E-Mini for the same degree and N. Thus the search ranges have to be different for each different tradable, N and degree.

To solve this problem and to have a standard search space for each tradable, I created a **Mult** input for each FadmV Velocity strategy and indicator .. If each tradable’s Velocity is multiplied by a number such that the standard deviation of that tradable’s Velocity is close to one standard deviation then the search space for vup and vdn for each tradable would be 0 to 3 standard deviations and we wouldn’t have to change the optimization search space every time we wanted to examine a new stock or future. The complicated equations that I use to normalize the ranges to one standard deviation for degree and N were derived using the software TableCurve 3D, automated surface and equation discovery. These equations are independent of the tradable and only depend on the polynomial degree and N. If you do not wish to use these equations then set iNorm=0. See note below .

How to Find The Mult.

Here is how to use the #**FadmVMultStd**. First load your chart with as much data as you are going to use. *Click* on **Insert-> Indicator**. Select the #**FadmVMultStd.** Modify the inputs to the indicator or just leave them at their default values. A table will be produced as shown below.

```
ES 1 min bars Date Range 1120501 to 1150501
Total Number of Bars=304156 Norm=1
Trading Times Constraint Start Time=0 EndTime=0
FadmVn Multiplier to Scale Velocity pw and N Range to One Std
```

pw	N	Std	1/Std	Std*Mult
1	20	0.351595	2.84418	0.988397
1	30	0.356036	2.80871	1.00088
1	40	0.360451	2.7743	1.01329
1	50	0.364442	2.72622	1.02451
1	60	0.368006	2.71735	1.03453
1	70	0.371198	2.69398	1.04351
		pw=1 Ave	2.76374	
2	20	0.34978	2.85894	0.983297
2	30	0.352419	2.83753	0.990715
2	40	0.355826	2.81036	1.00029
2	50	0.359166	2.78423	1.00968
2	60	0.362289	2.76023	1.01846
2	70	0.365198	2.73824	1.02664
		pw=2 Ave	2.79825	
3	20	0.348198	2.87193	0.978848
3	30	0.349252	2.86327	0.98181
3	40	0.351728	2.84311	0.98877
3	50	0.354446	2.82131	0.996412
3	60	0.357074	2.80054	1.0038
3	70	0.359576	2.78105	1.01083
		pw=3 Ave	2.8302	
4	20	0.34757	2.87712	0.977083

4	30	0.347231	2.87993	0.97613
4	40	0.348855	2.86652	0.980694
4	50	0.351063	2.84849	0.986903
4	60	0.353331	2.83021	0.993277
4	70	0.3555	2.81294	0.999376
		pw=4 Ave	2.85253	
		Mult=	2.81118	

For each tradable chart:

1. Apply the **FadmVMultStd** indicator to your chart.
2. Find the Multiplier as shown above.
3. Record the multiplier value and delete the indicator from your chart.
4. **Apply that multiplier to the Mult input** with **iNorm=1** in the strategy.
5. Do the parameter search for vup & vdn in the default range 0.25 to 3 by 0.25 .

Added Note:

The complicated equations that I use to normalize the ranges to one standard deviation were derived using the software TableCurve 3D, automated surface and equation discovery. These equations are independent of the tradable and only depend on the polynomial degree and N. If you do not wish to use these equations then set inorm=0. Then when you run **FadmVMultStd with inorm =0** you will get a multiplier that gives an average of all the ranges. With iNorm=0 you will have to determine the vup -vdn ranges for your optimization. They will be different for different tradables and time bars. Below is the output for iNorm=0 for **FadmVMultStd.** As you can see the variation for degree(pw)=2 is 6 times the variation for degree=4. And the variation for n-70 is five times the variation for N=20. so for this example in order to cover all degree and N ranges at 3 standard deviations maximum you would have to go from vup/vdn from 0.0005 to 0.09. Of course the best solution for iNorm=0 would be to modify the strategy multiplier for each degree and N which is essentially what my equation does but not as exact.

```
ES 1 min bars Date Range 1120501 to 1150501
Total Number of Bars=304156 Norm=0
Trading Times Constraint Start Time=0 EndTime=0
FadmVn Multiplier to Scale Velocity pw and N Range to One Std
```

pw	N	Std	1/Std	Std*Mult
2	20	0.00480974	207.912	1.82314
2	30	0.00259815	384.889	0.984833
2	40	0.00168332	594.062	0.638066
2	50	0.00120375	830.734	0.456284
2	60	0.000915983	1091.72	0.347204
2	70	0.000727482	1374.6	0.275753
		pw=2 Ave	747.321	
3	20	0.0139613	71.6267	5.29203
3	30	0.0075431	132.572	2.85922
3	40	0.00488588	204.671	1.852
3	50	0.00349297	286.289	1.32401
3	60	0.00265688	376.381	1.00709
3	70	0.00210918	474.118	0.799487
		pw=3 Ave	257.61	
4	20	0.0271776	36.795	10.3017
4	30	0.0146963	68.0445	5.57064
4	40	0.00951941	105.049	3.60834
4	50	0.00680606	146.928	2.57984
4	60	0.00517708	193.159	1.96238
4	70	0.0041091	243.362	1.55756
		pw=4 Ave	132.223	
		Mult=	379.051	