Contents

Part one

Predicting price changes with Signed Order Imbalance for equity securities

Objective:

This project aims to improve one period ahead price prediction using order imbalance information from the current time period.

Background information:

Motivation behind the project:

In this paper, the team dabbled at building one simple model based on Signed Order Imbalances (SOI), an idea inspired by Easley, Lopez de Prado and O'Hara's paper on "Flow Toxicity and Liquidity in a High Frequency World". Intuitively, SOI measures the magnitude of imbalance between buy and sell orders within a specified time period. And we believe a large portion of the short term equity price change is driven by such imbalances.

We restrict the range of this measure to be between 1 and -1. An SOI of 1 would indicate all orders are originated from buy requests, where as an SOI of -1 would indicate the opposite, and numbers in between quantify general imbalances with positive figures signaling preponderant buy orders, and vice versa.

Definition of SOI:

$$SOI = \sum_{i=1}^{t} w(i) * b(i)\,, b(i) \in \{0,1,-1\},\ \ w(i) = volume(i, i-1) / \sum_{k=1}^{t} volume(k, k-1) \qquad \textbf{(1)}$$

Explanation of the definition:

$\sum_{k=1}^{t} volume(k, k-1)$ defines the total volume of a bucket. A bucket is nothing more than an aggregation of adjacent smaller chunks of trades, known as bin. SOI is measured at the bucket level, and individual weighted bins contribute to the SOI measure over the bucket. There are numerous different ways of aggregating these bins into a bucket, and depending on the total trading volume of the specific stock at interest, the optimal parameters used in aggregation differ. The below table listed ways of aggregating trades into bins and buckets based on either time interval or volume interval, note tick data for bin choice implies using the each trade transaction directly as a bin unit without doing any aggregation:

Bin Choice	Bucket Choice
Time	Time
Trade Volume	Trade Volume
Tick Data	

A mixing of volume and time aggregating technique can be applied separately to bins and volume. In this paper, we primarily explore two different approaches: 1) tick data + time bucket, 2) tick data + volume bucket. While any combination of the above can be function, it is important to ensure a single bucket always contain at least one bin in order for the above methodology to be rational. And once the units are chosen, they should be fixed throughout the day. The advantage of using bins as oppose to tick data can be found in a paper written by Easley, López de Prado, and O'Hara [2012]. In their research, they claimed trade by trade classification is likely to result in misclassification (PG 22). However, ELO did not provide concrete evidence, but merely an intuitive suggestion. Without sufficient evidence that trade by trade classification is inferior, we choose to proceed with trade by trade (tick data) classification, and we will show trade by trade classification actually produces better regression fit than aggregating by time bins. Also by using tick data, we avoid unnecessary task of optimizing the fixed bin size within each bucket.

In equation (1), b(i) is a discrete function only takes value {+1,-1, 0}, which indicates the act (Bull/Sell/Neutral) of the corresponding bin/trade.

$$b(i) = \begin{cases} +1, & ith\ bin\ classfied\ as\ buy \\ 0, & ith\ bin\ classified\ as\ \ neutual \\ -1, & ith\ bin\ classified\ as\ sell \end{cases}$$

The classification method is explained in detail in the following section.

Buy/Sell/Neutral Classification:

Multiple buy/sell/neutral classification schemes are explored:

1) Modified "Lee-Ready" type classification based solely on trade data.

   In this algorithm, transactional level prices for trades are used. A trade is classified as a buy trade if the transactional price at time t is higher than the price at t-1, and classified as a sell trade if the transactional price at time t is lower than the price at t-1. In case both transactional prices are identical, we look back one period (i.e. t-2 to t-1 period) and follow the same procedure. The rationale behind this is in case of the tie, we implicitly assume the momentum will continue. If the t-2 to t-1 period return is still 0, then we classify the bin of t-1 to t as neutral and return 0. (I.e. this particular trade has no weight in calculating signed order imbalance)

Note our classification algorithm is inherently different from Lee-Ready's, because it skips the first step in the Lee-Ready algorithm, which is to first compare the transaction price with the nearest available quote midpoint, and only proceed to trades comparison if the price falls in the middle of the quote midpoint, whereas in our algorithm we only utilize trades.

Interested readers are encouraged to refer to Lee and Ready's research paper (Lee and Ready [1991])

2a) Nearest quotes with stochastic delays of 5 seconds.

For each trade, this algorithm looks at the nearest available quote ahead of it (in terms of exchange time). If the transacted price is closer to the bid of the quote, we classify the trade as a sell by setting b to -1 (i.e. we are taking the bid). Alternatively if the transacted price is closer to the ask of the quote, the trade is classified as a buy (or b = 1). If the transacted price is equidistant, we assigned the trade as neutral.
The trades reported to the consolidated tape are usually later than the reported quotes. In order to remedy the problem, we introduce stochastic delays to the quotes and follow the same algorithm for classification. Interested readers can refer to Lee-Ready's paper (Lee and Ready [1991]), in which they discussed about the optimal delay to be set at 5 seconds.

2b) Time weighted EMA quotes classification.

The third classification scheme first involves calculating time weighted EMAs of all the quotes, and starting with the nearest EMA quote, technique 2) is then applied, with the only change being the usage of EMA quotes instead of regular quotes.

The EMA calculation is defined as:

$$Q_{EMA(t)} = \frac{\sum_{i=0}^{t} weight(t_i) \times Q_{Mid(t_i)}}{\sum_{i=0}^{t} weight(t_i)} \text{ }, t \in [0, T] \qquad \textbf{(2)}$$

$$weight(t) = e^{(decay \times \sum \Delta t)}, t \in [0, T] \qquad \textbf{(3)}$$

3) Classic Lee-Ready with and without delay.

4) Classic Lee-Ready with and without delay using EMA mid quotes.

Tick Data with Time bucketing Analysis:

Before proceeding to building an actual predictive model, it is important to first ascertain our intuition that SOI is well correlated with price returns. The simple linear regression model we used to verify is as follows:

$$PReturn(k) = \alpha + \beta \cdot SOI_{tick,time}(k) + \varepsilon(k), \tag{4}$$

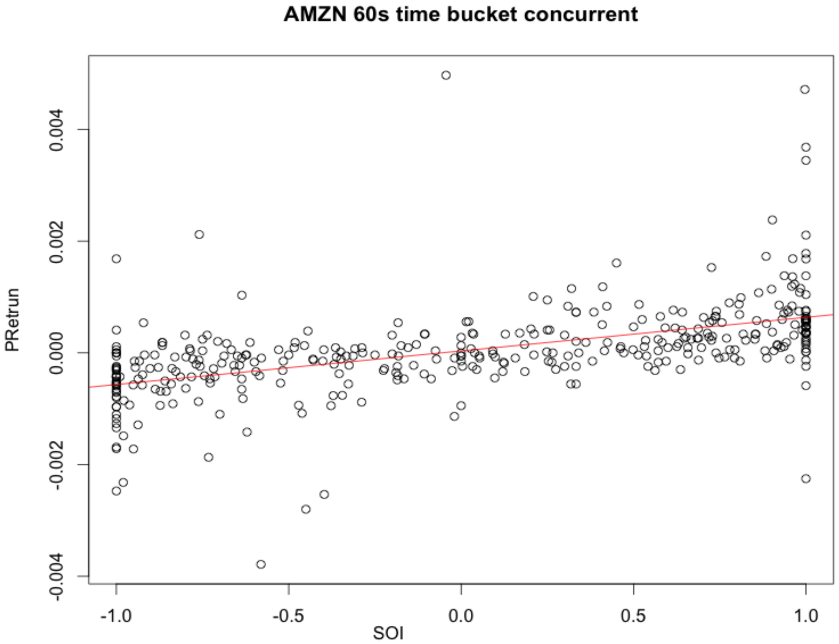
$$PReturn(k) = \log(\frac{last\ Quotes_{mid}\ in\ the\ kth\ bucket}{last\ Quotes_{mid}\ in\ the\ (k-1)th\ bucket}) \tag{5}$$

Here we define $SOI_{tick,time}$ using tick level data (i.e. each bin contains only one trade), and time buckets. The size of the time bucket is dependent on the average trading volume of the stock of interest:

$$time\ bucket\ (s) = \max(30s, \frac{avg(AMZN\ daily\ vol)}{avg(daily\ vol\ of\ stock\ of\ interest)} \times 60s) \tag{6}$$

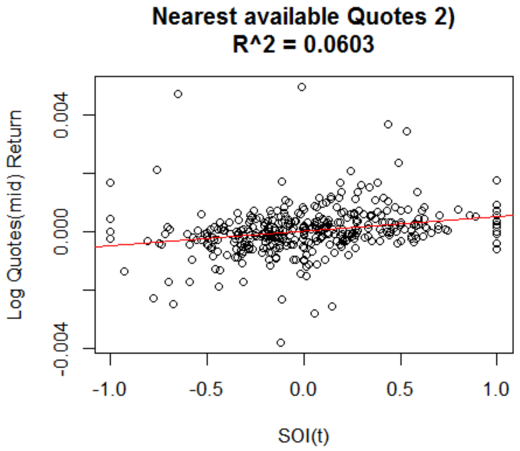
In the above equation, we used AMZN stock as an anchor; AMZN stock will have a time bucket of 60s. For other stocks with less trading volumes, their time buckets will be larger, for we would need to aggregate more data points. Since we are using linear scaling, we also want to cap the minimum bucket size to a constant to avoid bucket overflows.

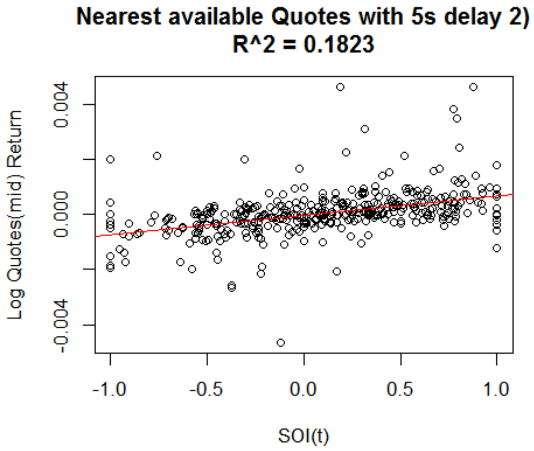
To help establishing feasibility, we tested AMZN stock for 4/23/2013, ran the simple regression model, and apply classification rule 1. Below plot is the regression outcome:

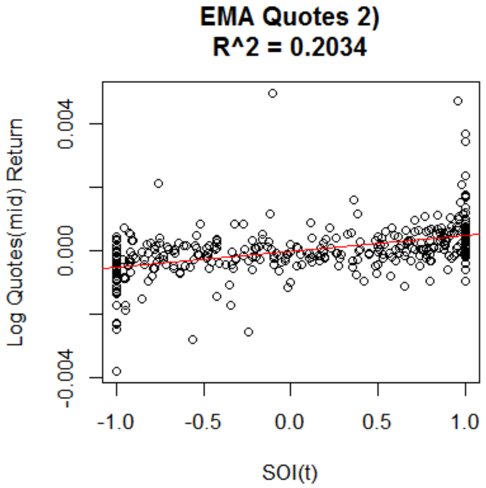


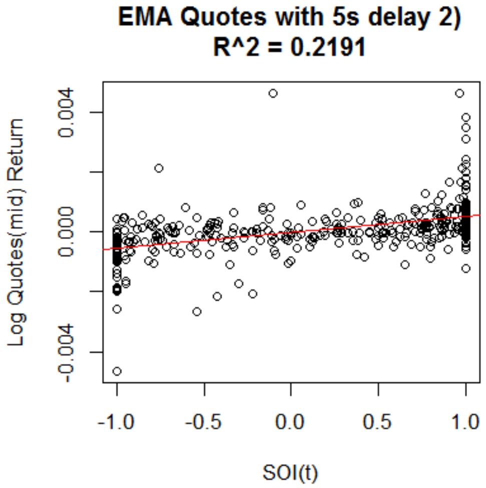
PRetrun(k) = $\alpha+\beta*SOI(k)+\varepsilon(k)$, R^2 = 27%

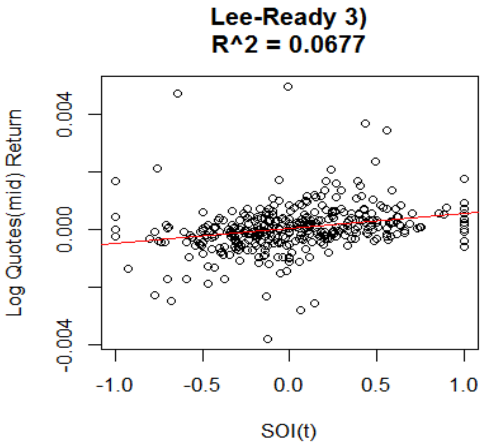
It is clear from the above plot SOI is strongly correlated with PReturn defined as mid-quote bucket return. Next we show classification rule #1 is superior to all the other rules that classify trades based on the midpoint of the nearest quotes, but we end up confirming Lee-Ready's 1991 paper, which stated 5s delay is ideal for quotes. Below we show the regression results with only changes in the classification rules employed:

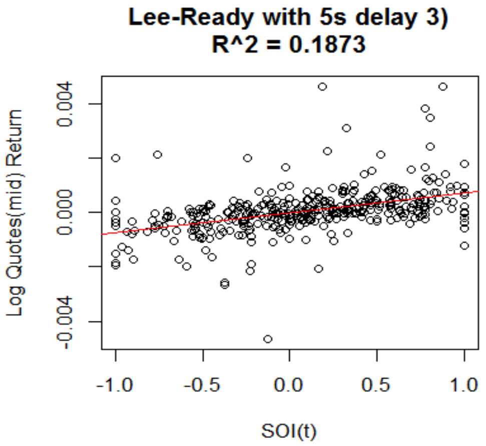

Nearest available Quotes 2)
R^2 = 0.0603

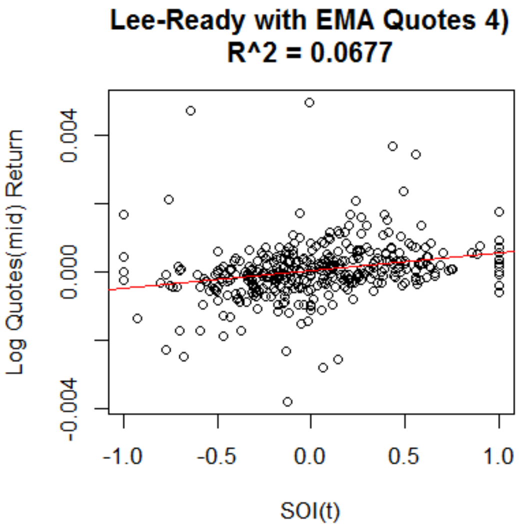

Nearest available Quotes with 5s delay 2)
R^2 = 0.1823

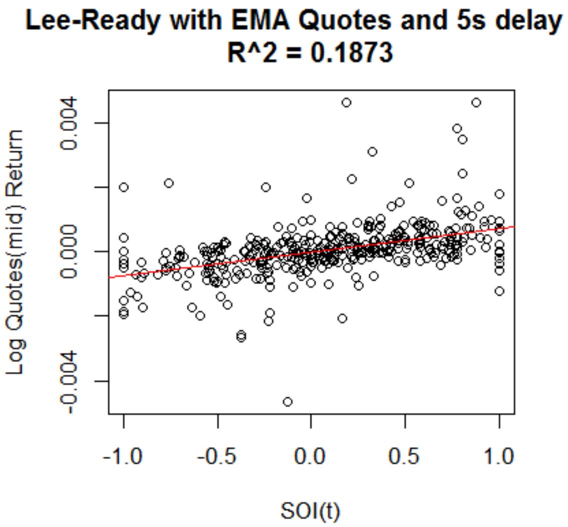

EMA Quotes 2)
R^2 = 0.2034


EMA Quotes with 5s delay 2)
R^2 = 0.2191


Lee-Ready 3)
R^2 = 0.0677

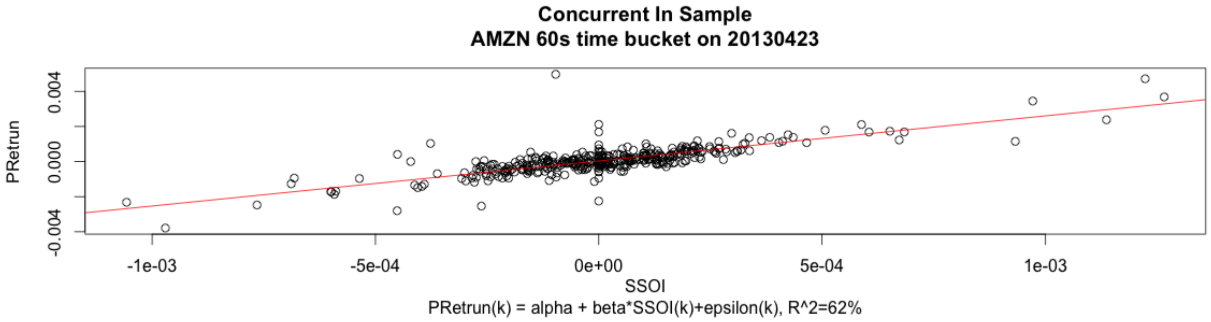

Lee-Ready with 5s delay 3)
R^2 = 0.1873

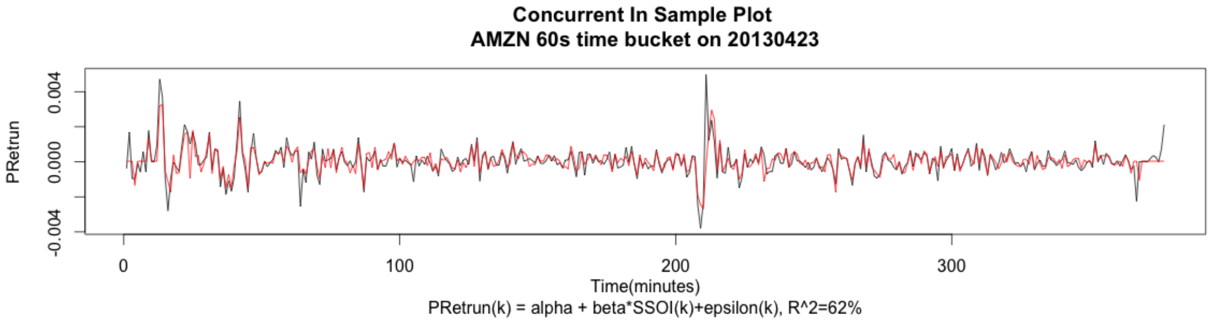




Classification rule #1 serves our purpose well. Note it is very likely all these classification methodologies that incorporates quotes delay of 5 seconds are likely to generate similar results; but since we were content with regression result, we moved on to improve this existing model. We were able to further adjust our model by scaling the SOI. The scaled SOI (SSOI) is obtained by multiplying SOI with the trade price volatility. And the price volatility of a bucket is defined as the standard deviation of all the trade prices within same bucket.

$$SSOI(k) = PVol(k) \cdot SOI(k) \tag{7}$$





We then tested the model robustness across time by assessing the beta stability. And 4/29/2013's AMZN data is used as an out of sample testing set.

Using the same SSOI sensitivity (beta), the concurrent regression model captures the price movement direction well, and correctly predicts 94.2% of the price movement direction on 29th. This ratio is calculated as $\frac{\sum \mathbb{I}\{sign(\widehat{PReturn_i}) = sign(PReturn_i)\}}{total\ \#\ of\ buckets}$. Therefore, we can conclude the SSOI sensitivity coefficient is relatively stable across days.

This model can also be successfully applied to any stock, and in Appendix I, we include the same concurrent model regression results for a list of different stocks across different sectors. The results corroborate to the conclusion that SOI scaled by volatility has a great explanatory power over mid quote returns.

Next we explore how to use this information to predict mid quote price returns. And the equation below summarizes our prediction model:

$$PReturn(k+1) = \alpha + \beta_1 \cdot SSOI_{tick,t}(k) + \beta_2 \cdot SSOI_{tick,s>\frac{t}{2}}(k) + \beta_3 \cdot PReturn(k) + \varepsilon(k) \qquad \textbf{(8)}$$

where $SSOI_{tick,s>\frac{t}{2}}$ measures partial SSOI, which only contains information for the second half of a time bucket.

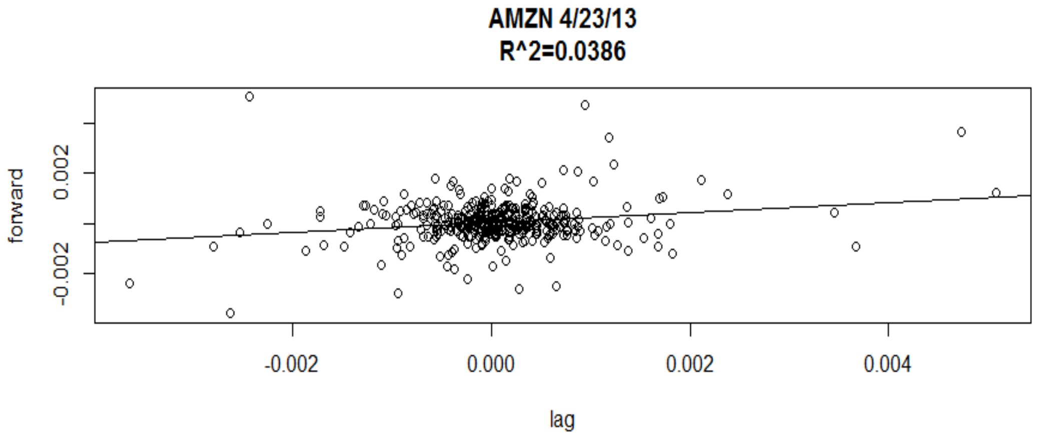
And we compare this model (equation (8)) with a simple autoregressive model:

$$PReturn(k+1) = \alpha + \beta \cdot PReturn(k) + \varepsilon(k) \qquad \textbf{(9)}$$

The reason we even consider an autoregressive model is because our initial concurrent plots show good correlation between price movement and SSOI. And one important conclusion we can reach from the previous observation about predictive model based on SSOI is that such a model implicitly makes the assumption that there exists autocorrelation within the price changes! And the value SSOI can add is to make the price prediction more accurate, assuming there exists autocorrelation among the quote prices. To provide an example, we again use AMZN stock for 4/23/13 with 60 seconds time bucket.

Here are the results:

If we use only the lag return, the model R



The R^2 = 3.86%, with a p-value = 4.08e-5

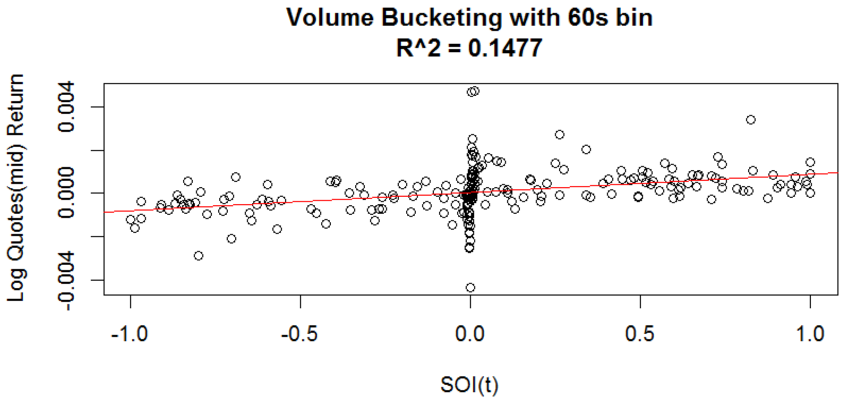
After employing model **(8)**, the R^2 increases to 6.42%, with an adjusted R^2 of 5.6%, and a p-value = 4.084e-5. We expand the analysis to multiple stocks for six different days, and we record adjusted R^2 improvement resulted from using our model against R^2 obtained using simple auto-regression. (See Appendix I for details)

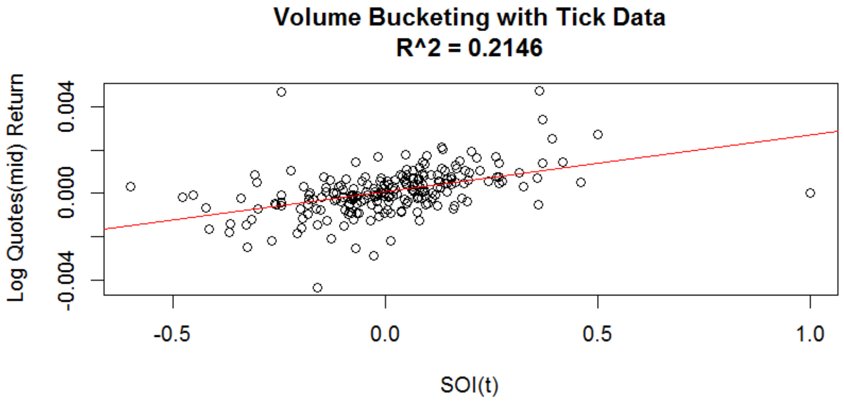
The results are satisfactory. For p-values < 0.1, the model achieves higher adjusted R^2 in 82.5% of the cases. And for p-values <0.05, this model achieves higher adjusted R^2 in 90.91% of the cases. These results suggest when there is autocorrelation present in mid quote returns, adding SSOI as additional independent variables can improve the performance of our predictive model.

Tick Data with Volume bucketing Analysis:

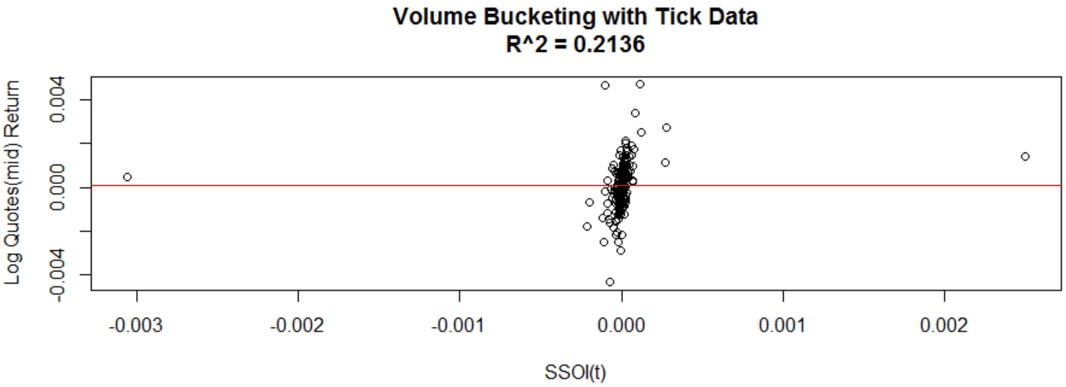
In this section, we strive to examine the effect of using time bin instead of tick data; and we explore the adjustment we need to make to the previous predictive model and what are the essential differences between the two bucketing mechanisms.

First, we compare the difference in using time bins and tick data. We have again used 4/23/2013's AMZN data as a testing set. Unlike ELO has suggested that using trade-by-trade classification can lead to great misclassification, trade-by trade classification clearly is a winner here:


Volume Bucketing with 60s bin
R^2 = 0.1477


Volume Bucketing with Tick Data
R^2 = 0.2146

Another phenomenon we have noticed is scaling the SOI measure by a volatility measure does not increase model fitting accuracies, the following plot is a clear demonstration:



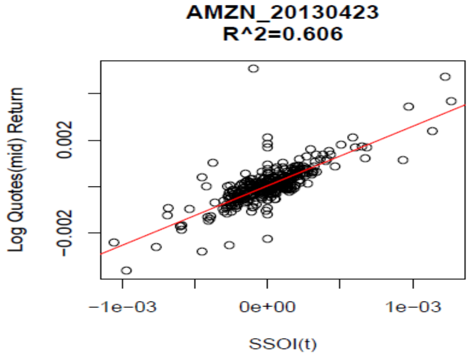
And for the prediction model, we modify equation **(8)** to be

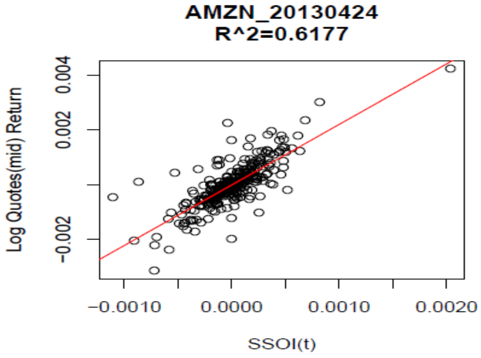
$$PReturn(k+1) = \alpha + \beta_1 \cdot SOI_{tick,v}(k) + \beta_2 \cdot SOI_{tick,u>\frac{v}{2}}(k) + \beta_3 \cdot PReturn(k) + \varepsilon(k) \quad \textbf{(10)}$$

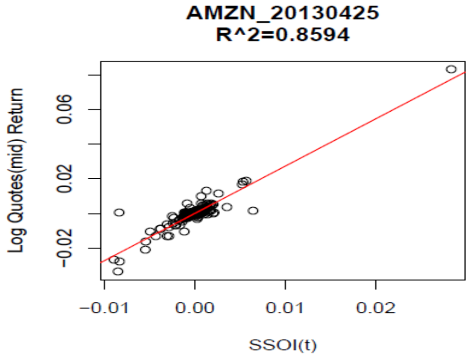
And we repeat the same procedure. This time, for p-value < 0.1, the predictive model beats the autoregressive model only 46.67% of the times, and for p-value < 0.05, this number drops to 40%. (for details, please check Appendix I). Yet we have also noticed the general drop in the % of p-values < 0.1. This is what is happening: by using volume bucketing, we are creating unequally spaced time forecasting intervals (unless trades occur uniformly throughout the day, and this is clearly false). Therefore, it is not realistic to employ the same autocorrelation model as a benchmark. Yet, if lag bucket mid quote returns no longer have any predictive power, then the same model based on SOI falls apart as well. This is the conclusion we have reached in the previous section.

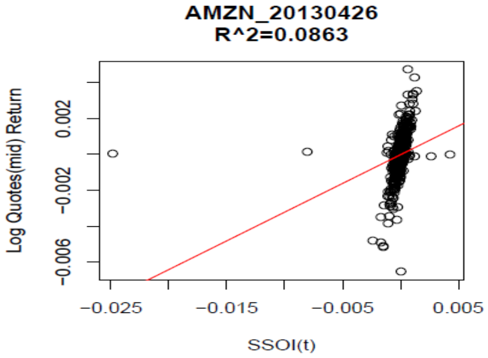
Therefore, we conclude Tick Data/time bin with time bucketing should be the ideal way to construct SOI, which can be further utilized to improve an existing price change prediction model, given we can observe autocorrelation among short-term lags.

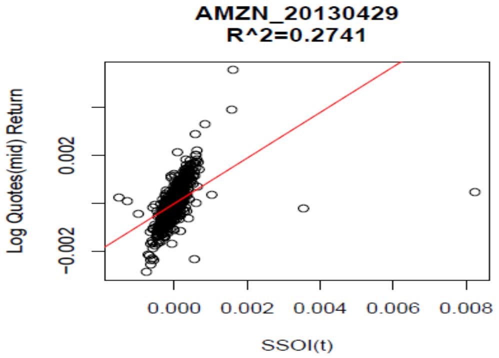
APPENDIX I:

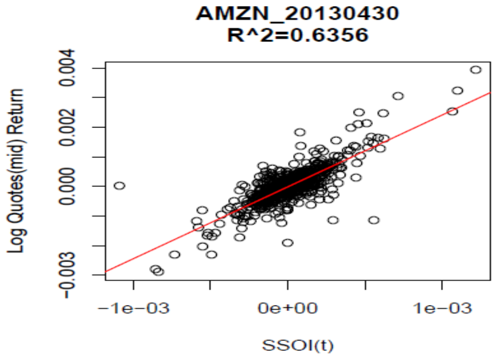

AMZN_20130423
R^2=0.606
Log Quotes(mid) Return
−0.002
0.002
−1e−03
0e+00
1e−03
SSOI(t)

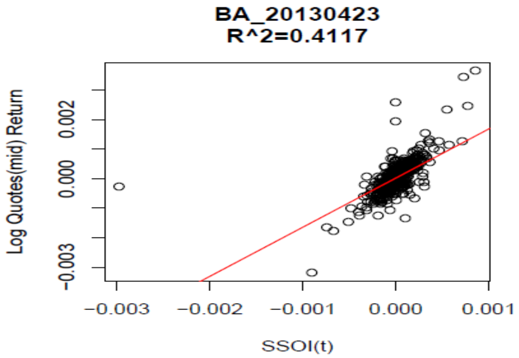

AMZN_20130424
R^2=0.6177
Log Quotes(mid) Return
−0.002
0.002
0.004
−0.0010
0.0000
0.0010
0.0020
SSOI(t)

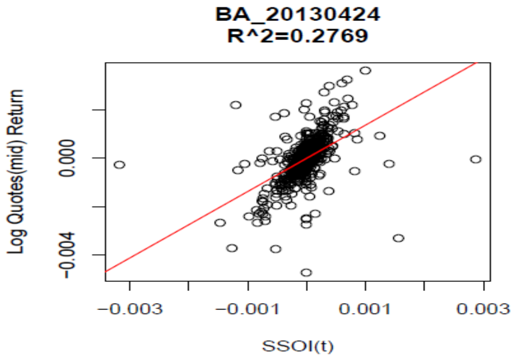

AMZN_20130425
R^2=0.8594
Log Quotes(mid) Return
−0.02
0.02
0.06
−0.01
0.00
0.01
0.02
SSOI(t)


AMZN_20130426
R^2=0.0863
Log Quotes(mid) Return
−0.006
−0.002
0.002
−0.025
−0.015
−0.005
0.005
SSOI(t)

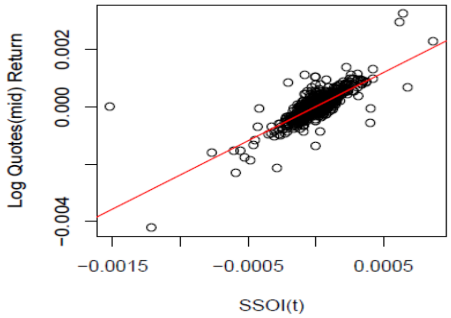

AMZN_20130429
R^2=0.2741
Log Quotes(mid) Return
−0.002
0.002
0.000
0.002
0.004
0.006
0.008
SSOI(t)


AMZN_20130430
R^2=0.6356
Log Quotes(mid) Return
−0.003
0.000
0.002
0.004
−1e−03
0e+00
1e−03
SSOI(t)

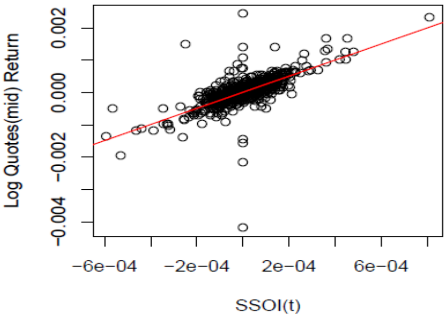

BA_20130423
R^2=0.4117
Log Quotes(mid) Return
−0.003
0.000
0.002
−0.003
−0.002
−0.001
0.000
0.001
SSOI(t)


BA_20130424
R^2=0.2769
Log Quotes(mid) Return
−0.004
0.000
−0.003
−0.001
0.001
0.003
SSOI(t)

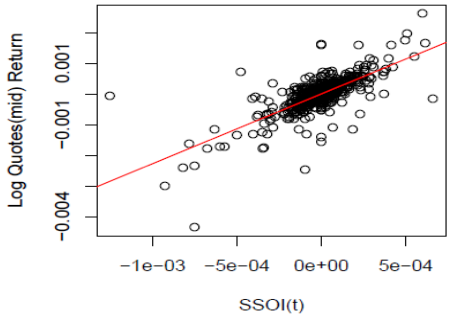
BA_20130425
R^2=0.6161



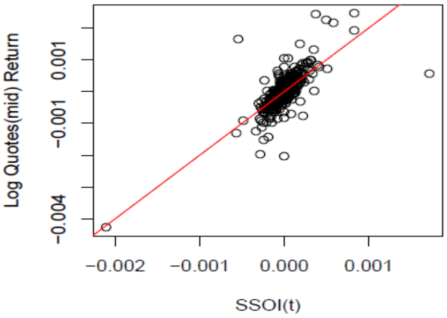
BA_20130426
R^2=0.4917



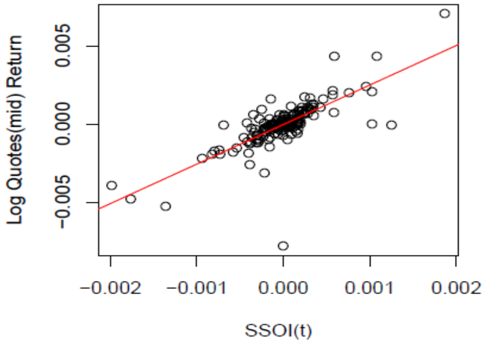
BA_20130429
R^2=0.5469



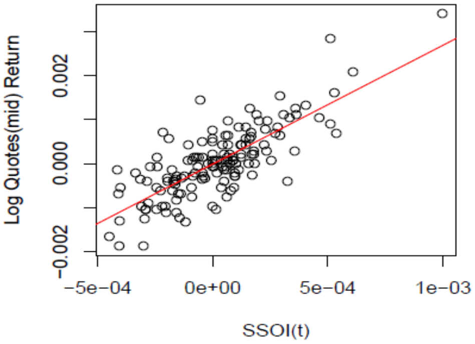
BA_20130430
R^2=0.5396



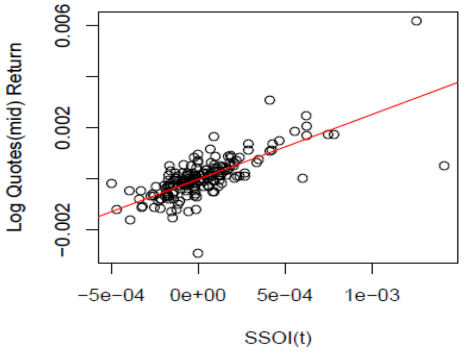
TIF_20130423
R^2=0.5738



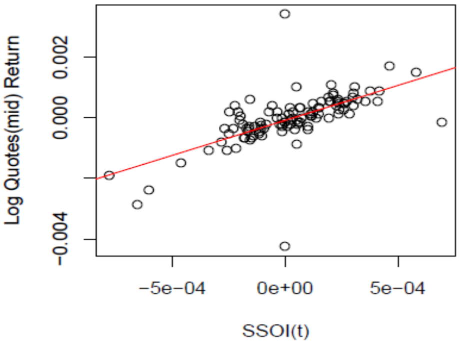
TIF_20130424
R^2=0.6071

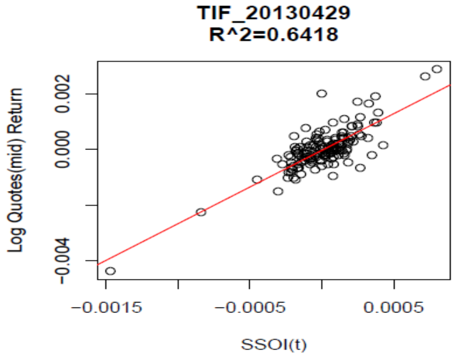


TIF_20130425
R^2=0.5369

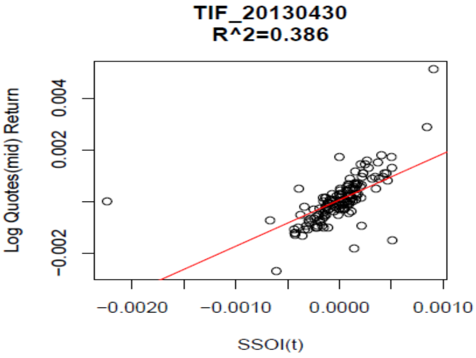


TIF_20130426
R^2=0.4047

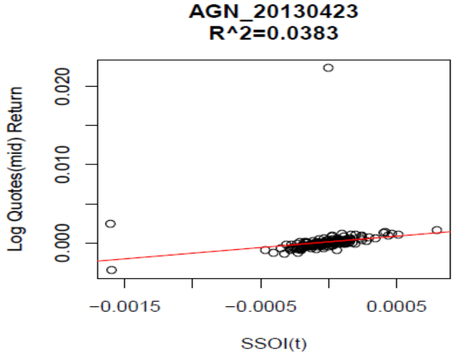




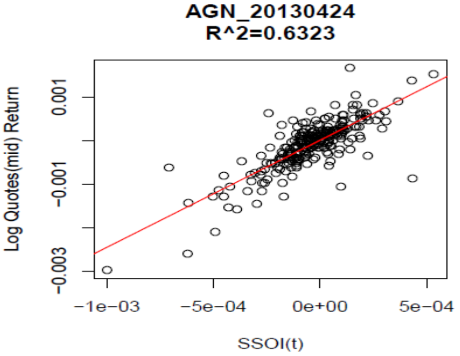
TIF_20130429
R^2=0.6418



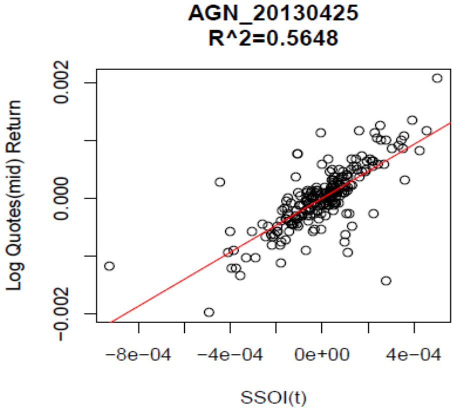
TIF_20130430
R^2=0.386



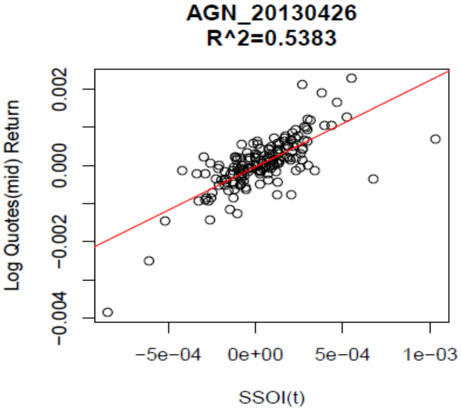
AGN_20130423
R^2=0.0383



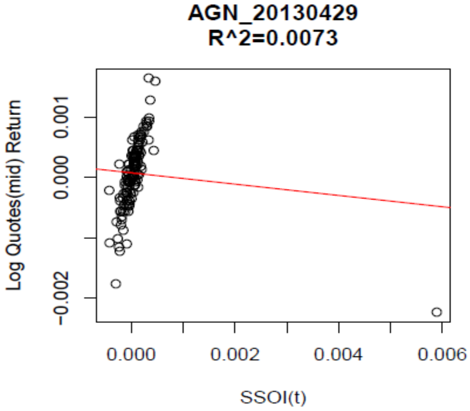
AGN_20130424
R^2=0.6323



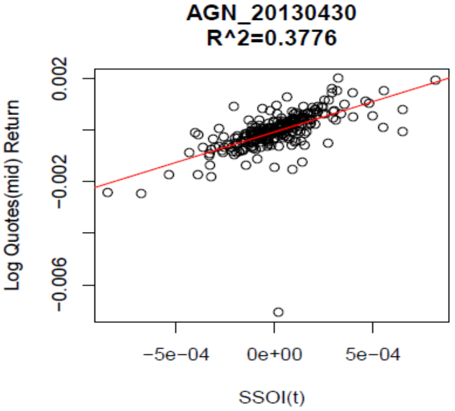
AGN_20130425
R^2=0.5648



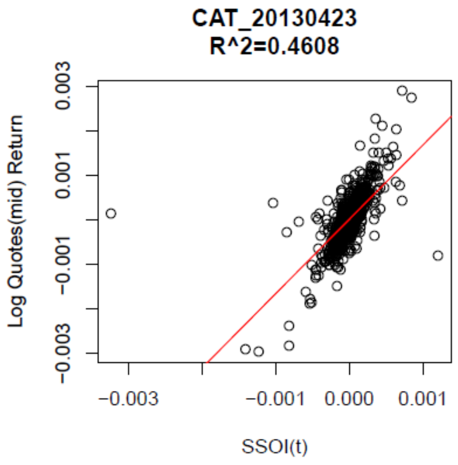
AGN_20130426
R^2=0.5383

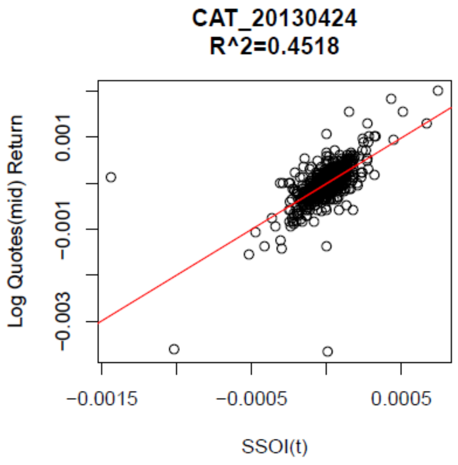


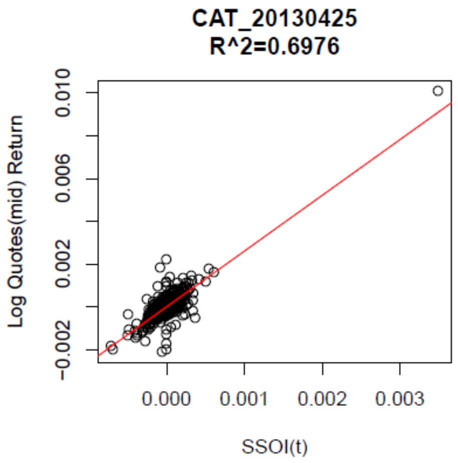
AGN_20130429
R^2=0.0073

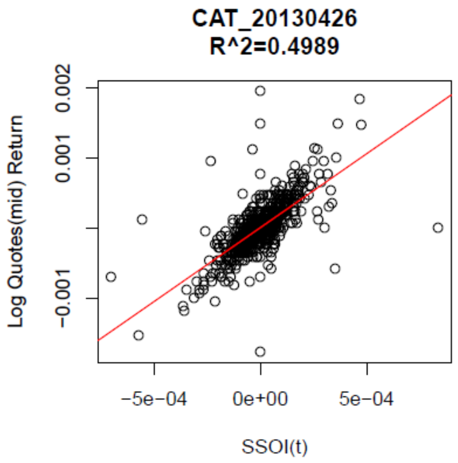


AGN_20130430
R^2=0.3776









Time Bucketing Predictive Model

Stock	Date	R^2 (SSOI)	Adj R^2 (SSOI)	R^2 (Lagged Reg)	p-value
AMZN	20130423	6.42%	5.60%	3.86%	2.87E-05
	20130424	0.50%	-0.57%	0.47%	0.249844835
	20130425	7.98%	7.67%	7.19%	1.61E-16
	20130426	1.11%	0.97%	0.63%	0.000105969
	20130429	1.07%	0.79%	0.43%	0.031763481
	20130430	1.54%	1.05%	0.12%	0.290685702
BA	20130423	0.67%	0.02%	2.09%	0.107922369
	20130424	1.14%	0.88%	0.09%	0.36387652
	20130425	4.34%	3.98%	3.15%	7.91E-09
	20130426	0.87%	0.49%	0.41%	0.352066078
	20130429	1.65%	1.28%	0.49%	0.097129845
	20130430	1.61%	1.12%	1.60%	0.002936077

TIF	20130423	1.37%	-0.54%	0.31%	0.458155075
	20130424	3.96%	1.73%	2.46%	0.143392095
	20130425	3.15%	1.36%	7.65%	0.14720319
	20130426	0.57%	-2.33%	0.18%	0.600787765
	20130429	2.31%	0.29%	2.21%	0.07743394
	20130430	4.71%	2.48%	3.51%	0.069604588
AGN	20130423	0.35%	-1.57%	0.00%	0.995780155
	20130424	2.32%	1.13%	0.88%	0.084569109
	20130425	3.20%	1.84%	1.65%	0.052116863
	20130426	8.64%	6.82%	3.95%	0.005580231
	20130429	8.39%	6.38%	5.04%	0.006851876
	20130430	6.31%	5.19%	0.01%	0.857168256
CAT	20130423	0.34%	0.17%	0.00%	0.91573371
	20130424	0.52%	0.22%	0.68%	0.266328501
	20130425	2.29%	1.97%	0.66%	1.07E-05
	20130426	1.94%	1.58%	0.26%	6.70E-05
	20130429	1.88%	1.52%	1.05%	0.000536967
	20130430	6.88%	6.59%	0.07%	0.300202829

Volume Bucketing Predictive Model

Stock	Dates	R^2 (SOI)	Adj R^2 (SOI)	R^2 (lag)	p-value
AMZN	20130423	3.74%	2.40%	2.40%	0.021131
	20130424	0.27%	-1.45%	0.25%	0.507085
	20130425	12.81%	12.34%	11.84%	5.71E-17
	20130426	3.40%	3.19%	2.69%	8.19E-10
	20130429	0.17%	-0.25%	0.00%	0.943753
	20130430	0.25%	-0.48%	0.02%	0.783658
BA	20130423	0.88%	-0.08%	0.02%	0.819185
	20130424	0.60%	0.27%	0.05%	0.511789
	20130425	0.26%	-0.26%	0.05%	0.596868
	20130426	2.63%	2.14%	0.07%	0.526989
	20130429	3.66%	3.19%	3.43%	3.18E-06
	20130430	6.81%	6.21%	5.66%	1.60E-07
TIF	20130423	3.13%	0.28%	0.53%	0.458153
	20130424	3.00%	-0.20%	2.38%	0.13821
	20130425	12.06%	9.64%	5.56%	0.009915
	20130426	12.27%	8.02%	3.53%	0.119152

	20130429	1.74%	-1.36%	1.25%	0.274572
	20130430	4.38%	0.92%	3.91%	0.06913
AGN	20130423	0.65%	-2.25%	0.24%	0.61603
	20130424	0.45%	-1.33%	0.44%	0.390572
	20130425	0.78%	-1.58%	0.57%	0.397964
	20130426	7.41%	4.63%	0.69%	0.390141
	20130429	10.44%	7.24%	9.54%	0.003653
	20130430	1.28%	-0.55%	0.27%	0.503657
CAT	20130423	0.06%	-0.18%	0.02%	0.585076
	20130424	0.48%	0.05%	0.46%	0.072035
	20130425	1.25%	0.79%	1.04%	0.010152
	20130426	2.11%	1.59%	1.38%	0.00518
	20130429	4.00%	3.50%	3.87%	1.52E-06
	20130430	0.92%	0.49%	0.14%	0.332183