MAFS 6010Z Project 3
Cryptocurrency Trading

Price-Volume Factor: Tianying Zhou, Yijia, Ma, Langting Weng
CNN-ALSTM Model: Aoran Li

Abstract

This study proposes several innovative trading strategies for cryptocurrency markets. Each strategy is designed to exploit specific market conditions and maximize profit potential. Extensive backtesting and performance analysis are conducted to evaluate the effectiveness of each strategy. In the end, we design a model which combines CNN, LSTM, and Attention architecture,and explore its effectiveness in predicting price increases or decreases.

1 Overview of Data Structure

We initially conducted a simple visualization of the datasets, selecting the average price, i.e. $(open + high + low + close)/4$, and volume at 00:00 each day for plotting. The results are shown below. It is evident that there are significant differences in the magnitudes of price and volume for each currency, and they appear to be inversely proportional. Currencies with larger price magnitudes, such as BTC, tend to have smaller volumes, and vice versa. At the same time, we can observe that the price trends of the four currencies are very similar.

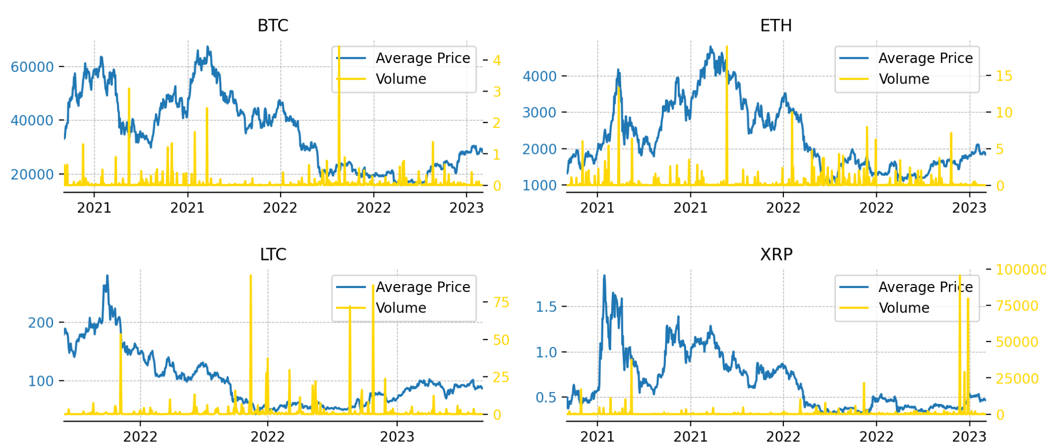

Figure 1: Historical Average Price and Volume

2 Price-Volume Factor

Based on the provided data (OHLCV), we initiate the analysis from the perspective of volume and price data. We categorize them into 4 major groups based on the intrinsic meaning of volume-price factors. Subsequently, we conduct single-factor tests on the volume-price factors within these 4 major groups. Ultimately, we have identified 7 factors that are relatively effective and exhibit strong logical coherence. The following table provides a summary of our factors.

Table 1: Price-Volume Factor Summary

Category	Name	Formula
Momentum	Golden Cross	$EWMA(Close, window1)$ $-EWMA(Close, window2)$
	Momentum Term Spread	$\frac{Close_t - Close_{t-window1}}{Close_{t-window1}} - \frac{Close_t - Close_{t-window2}}{Close_{t-window2}}$
Volatility	Volume Volatility	$-std(Volume)$
	Price Change Volatility	$abs(Open - Close) * std(Close)$
	Return Volatility	$\log(\frac{Open - Close}{Open})/std(Close)$
Price-Volume Divergence	Price-Volume Correlation	$-corr(Close, Volume, window)$
Long-Short Comparison	LS Comparison Change	$EWMA(\alpha, window1)$ $-EWMA(\alpha, window2)$ $\alpha = Volume \times \frac{(Close-Low)-(High-Close)}{High-Low}$

2.1 Golden Cross & Death Cross

When the short-term EWMA crosses above the long-term EWMA, forming a Golden Cross, it may be considered a buy signal, indicating a potential upward trend. Conversely, when the short-term EWMA crosses below the long-term EWMA, forming a Death Cross, it may be considered a sell signal, indicating a potential downward trend.

$$GC = EWMA(Close, window1) - EWMA(Close, window2), window1 > window2 \quad (1)$$

Selecting a trading frequency of 30 minutes, as seen from the balance curve in the chart, the strategy experienced a decline both at the beginning and the end of the 15-day backtest. There was a steady mid-term increase, but with some elevated volatility.

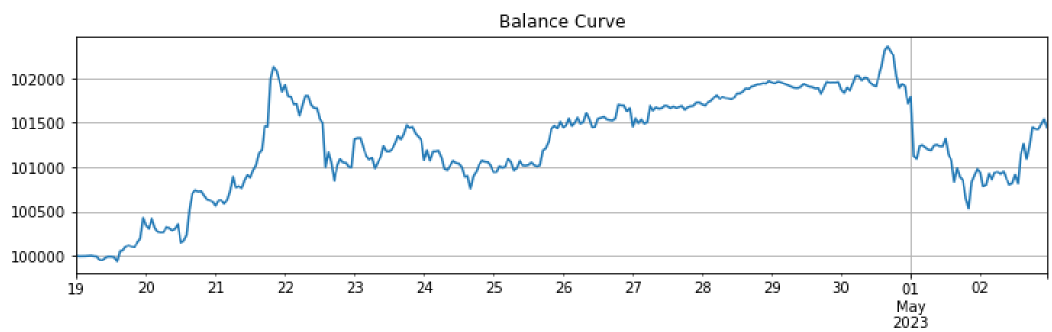

Figure 2: Balance curve of Golden Cross factor

Total Return	Average Daily Return	Sharpe Ratio	Maximum Drawdown
1.447127%	0.103953%	3.788373	-0.020199

2.2 Momentum Term Spread

Momentum Term Spread can filter short-term noise: Short-term momentum may be influenced by significant price fluctuations, potentially containing more short-term noise. Subtracting short-term momentum helps filter out some of this short-term volatility, allowing the factor to concentrate on longer-term trend changes. Here $window1 > window2$:

$$MTS = \frac{Close_t - Close_{t-window1}}{Close_{t-window1}} - \frac{Close_t - Close_{t-window2}}{Close_{t-window2}} \tag{2}$$

The strategy appears more robust from the balance curve, displaying a predominantly upward trend. Achieving a gain of 2500 within 15 days is already quite significant.

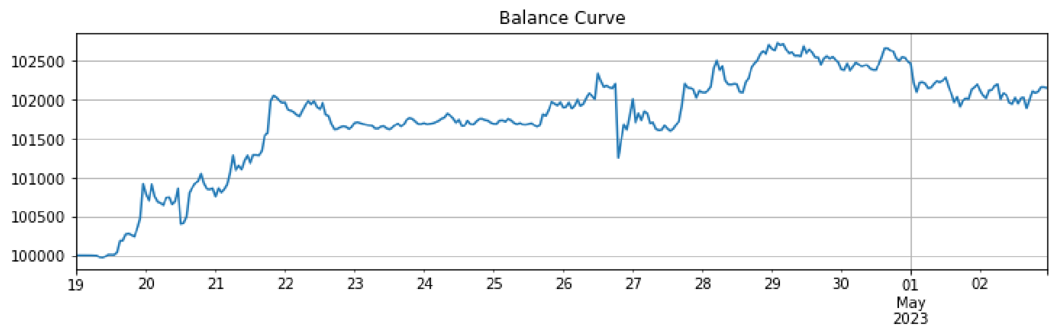

Figure 3: Balance curve of Momentum Term Spread factor

Total Return	Average Daily Return	Sharpe Ratio	Maximum Drawdown
2.150724%	0.152986%	6.713743	-0.013836

2.3 Volume Volatility

Trading volume volatility, defined as the negative of the standard deviation of trading volume over a certain period, represents an industry with stable bullish market sentiment. We opt to long with lower volatility and short with higher volatility.

$$VV = -std(Volume) \tag{3}$$

It can be observed that the results of this strategy bear some resemblance to the Golden Cross, both experiencing a decline before and after. However, it's evident that this strategy exhibits lower volatility. This is evident from the comparison of the Sharpe ratio and max drawdown.

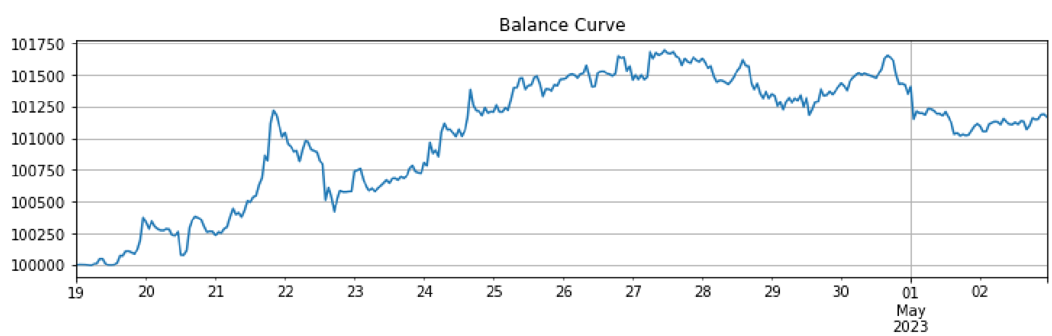

Figure 4: Balance curve of Volume Volatility factor

Total Return	Average Daily Return	Sharpe Ratio	Maximum Drawdown
1.164851%	0.083206%	5.104606	-0.008435

2.4 Price Change Volatility

Price change volatility(PCV) is defined as the open price minus the close price term multiple the standard deviation of the close price. The idea of this factor is to increase the performance of price volatility. We sell assets if the PCV has a rising trend and buy assets if the PCV has a downward trend, which means we long assets when the market is stable and short assets when the market is unstable. The PCV factor is calculated as:

$$PCV = abs(Open - Close) * std(Close) \tag{4}$$

Below is our backtest result via PCV strategy.

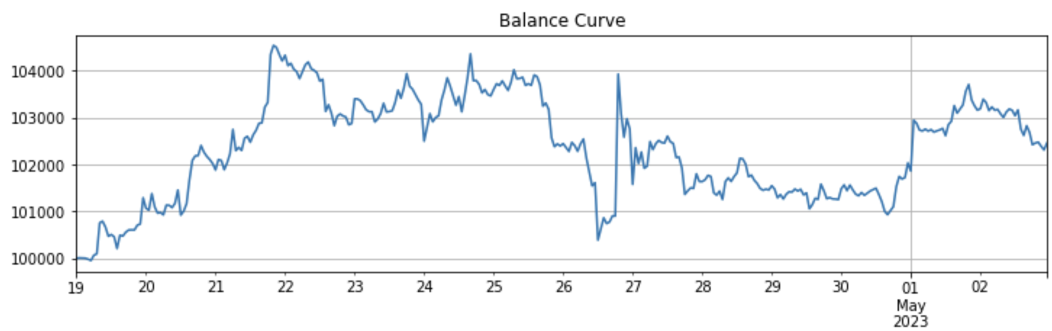

Figure 5: Balance curve of PCV factor

Total Return	Average Daily Return	Sharpe Ratio	Maximum Drawdown
2.450608%	0.177662%	3.412259	-0.040614

2.5 Return Volatility

Return volatility(RV) is defined as the price return divided by the standard deviation of the close price. The price return is calculated as the open price minus close price term divided by open price first, then takes the logarithm. The logic behind this factor is that we long position when the RV factor is smaller than the lower bound, and we short position when the RV factor is larger than the upper bound. The RV factor is calculated as:

$$RV = \log(\frac{Open - Close}{Open})/std(Close) \tag{5}$$

Below is our backtest result via RV strategy.

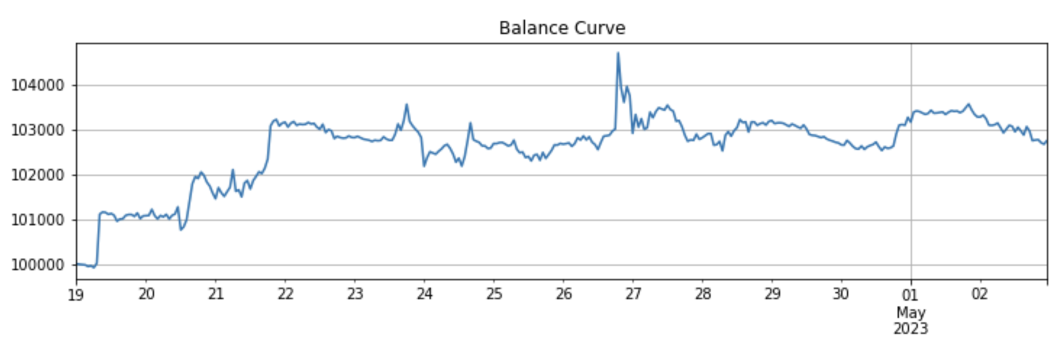

Figure 6: Balance curve of RV factor

Total Return	Average Daily Return	Sharpe Ratio	Maximum Drawdown
2.741925%	0.195626%	5.390716	-0.029291

2.6 Price-Volume Correlation

Price-volume divergence refers to a situation where, over a specified period, trading volume increases while prices decline, or trading volume decreases while prices rise. The higher the degree of price-volume divergence, the greater the probability of increased excess returns. Here, price-volume divergence is defined as the negative correlation coefficient between trading volume and closing price over a specified period($window = 90min$):

$$PVC = -corr(Close, Volume, window) \tag{6}$$

The choice of a relatively long trading frequency (bar_length) is made because if the time period is too short, price fluctuations may be too small, leading to a potential correlation coefficient of NaN.

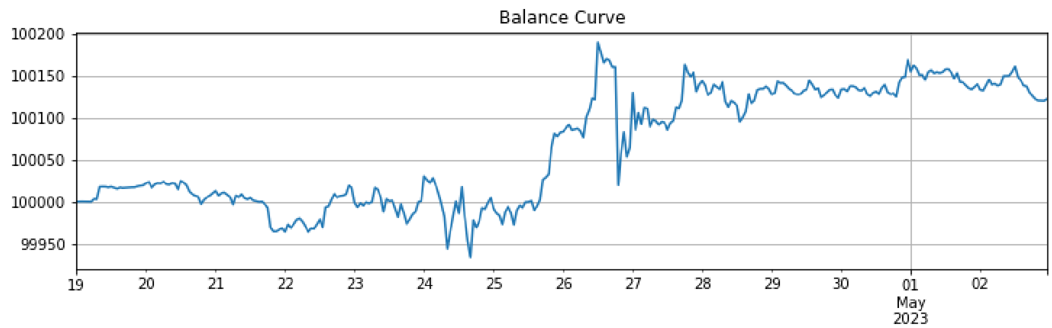

Figure 7: Balance curve of Price Volume Correlation factor

Total Return	Average Daily Return	Sharpe Ratio	Maximum Drawdown
0.122059%	0.872040%	4.343424	-0.002094

2.7 Long-short Comparison Change

Firstly, calculate the long-short power contrast. For the numerator, we subtract the bullish power from the bearish power, namely $(Close - Low) - (High - Close)$. The denominator is the difference between the highest and lowest prices, representing the extreme values of the intraday price range. Multiply the obtained long-short power contrast by the industry trading volume for the day, and the absolute value represents the daily long-short power contrast in monetary terms.

By taking the exponential weighted average of the long-short power contrast on a daily basis for the long term and subtracting the exponential weighted average for the short term, we can determine the recent change in the long-short power contrast relative to the mean of the long-term contrast. A larger factor value indicates a weakening of bullish relative to bearish power in the near term, while a smaller factor value suggests an increase in the contrast between long and short positions relative to the long-term. Here $window1 > window2$:

$$\begin{aligned} LSCC = \text{EWMA}\left(Volume \times \frac{(Close - Low) - (High - Close)}{High - Low}, window1\right) \\ - \text{EWMA}\left(Volume \times \frac{(Close - Low) - (High - Close)}{High - Low}, window2\right) \end{aligned}$$

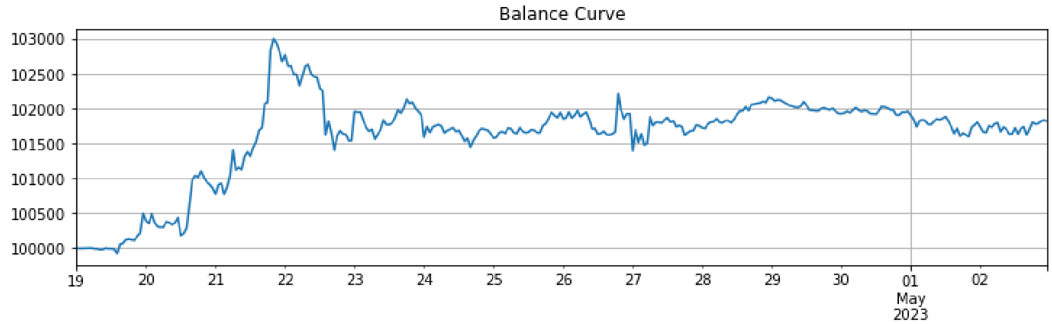

Figure 8: Balance curve of LSCC factor

Total Return	Average Daily Return	Sharpe Ratio	Maximum Drawdown
1.817761%	0.130577%	3.975008	-0.017485

3 CNN-LSTM-Attention Model

In this section, I mainly use deep learning models to predict the short-term future price of Bitcoin. I combine CNN, LSTM, and Attention modules to build a neural network model. You can check my code in My Github Repo

3.1 Data Preprocessing

The raw data is the OHLCV data of 4 Bitcoins at every minute. To ensure comparability between different currencies and features, and to facilitate model learning, I performed normalization on the original features, which is calculated as:

$$OHLCV_{per\ min}/mean_{previous\ day}(OHLCV)$$

This approach reduces the impact of data scale on the model. In addition, within the sample, the data ranges from 2021.9.14 to 2023.4.18, with a total of 838,079 pieces of data for each currency, spanning a total of 582 days. Therefore, to ensure the training effect of the model, I selected 2023.1.1 to 2023.4.18, as the model validation set, and the rest 2021.9.14 to 2023.1.1 as the model training set. Besides, I use future 5 min return as my predicted label, whose 1 stands for positive return and 0 stands for negative return.

The specific data parameters are shown below:

- Training period: 2021.9.14-2023.1.1
- Validation: 2023.1.1-2023.4.18
- Total Data length: 4 * 838079 = 3352316
- Input feature: O, H, L, C, V (normalized)
- Output Label: Up or Down (1/0)in future 5min

3.2 Model Construction

In the model structure design, I combined CNN, LSTM, and attention modules to build the model for training.

Due to the small instantaneous changes in the price features of OHLCV in the time series, which is similar to pixel data in image processing, CNN was considered for feature extraction.

After extracting temporal features using CNN, LSTM-attention module was used to capture the variations in the temporal dimension of the features.

Traditional LSTM only uses the last step of the last layer as output prediction, so I added the attention module, which allows all time steps of the last layer output by LSTM to be considered in the model, enabling the model to further capture features at all times. Finally, a fully connected layer maps the features into the output dimension for prediction.

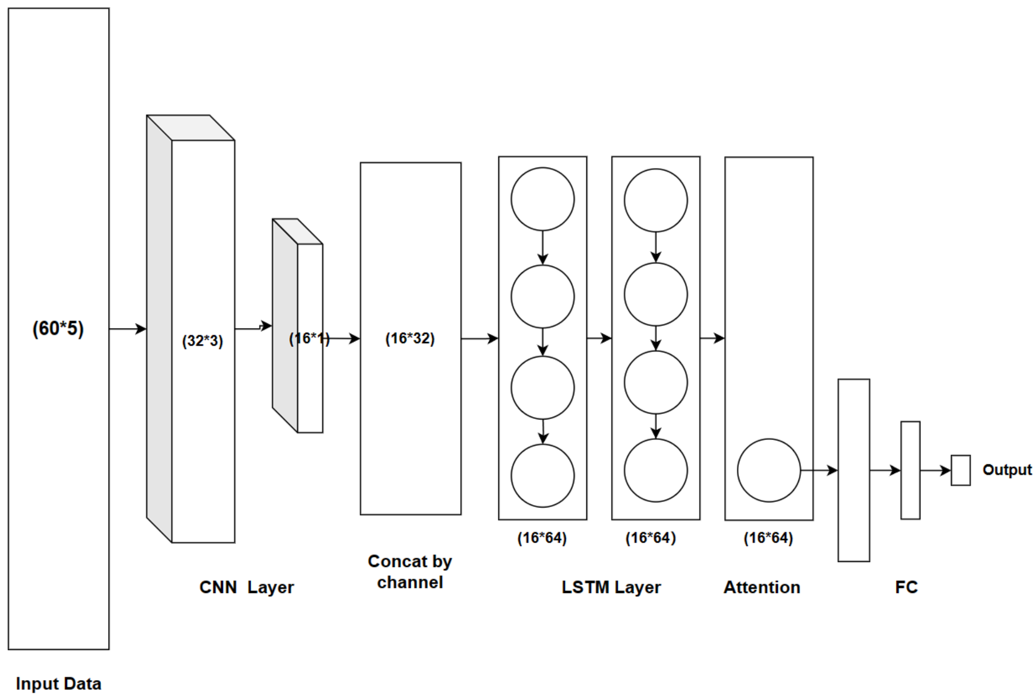

Figure 9: Model Architecture

- **Input Shape**

  In order to capture the changes in the sequence over a considerable period of time, I set the length of the sequence to 60, which is equivalent to 1 hour of data: (Sequence length, features): (60, 15)

- **CNN layer**

  In the CNN layer, I hope it can efficiently aggregate sequences across different time lengths, so I designed the kernel size to be (10, 2) and selected 16 and 32 channels to capture different features.

  After introducing two layers of CNN and a pooling layer, the data dimension is transformed into (32, 16, 1).

  Due to the characteristics of the CNN model, the 32 channels can be considered as different features, and the length of 16 can be considered as a time series. Therefore, in the output layer of the CNN, I transformed the data dimension and output it as (16, 32).

- **LSTM-Attention**

  In this section, I selected a LSTM model with num_layer$\bar{2}$ and added a self-attention mechanism to the last layer output of the LSTM in order to consider the information of the entire sequence.

  Finally, I selected the output at the last time step as the input to the fully connected layer.

- **Other Tricks**

  I used cross-entropy loss in my model. And I also used early-stopping, decayed-linear rate, balanced label sampling and some other tricks in my model's training.

3.3 Trading Strategy Design

In this section, I designed a specific trading strategy based on the output of my model:

- In order to reduce the impact of my orders on market prices, I selected the average volume of the past 60 minutes as the benchmark.

- I give different order quantities based on the strength of the model signal: when the probability of the model signal is greater than 0.9, I will place an order that is twice the benchmark; when the probability of the model signal is greater than 0.5, I will place an order that is 0.5 times the benchmark.

- To control my risk, I will only maintain each position for 5 minutes, which means that I will close positions opened 5 minutes ago every minute.

- At the same time, in order to reduce the impact of extreme situations on my positions, when my model outputs three consecutive upward (downward) signals, which means that there may be extreme upward (downward) market conditions in the future, I will close down positions opened in the past 5 minutes (i.e. close positions in the opposite direction of the signal) to control risks.

- Finally, in order to maintain my cash flow, when my cash is less than a certain amount, I will stop opening long positions and only place orders in the downward direction.

3.4 Model Results & Future Works

Unfortunately, due to time, computational resources and personal reasons, I did not fully train and save the entire model for backtesting.

Loss	0.69
Acc	0.523
F1	0.362

Table 2: Model Training Performance

Currently, only the performance of the loss function and accuracy of the model on the training set and validation set are available. It can be seen that the accuracy of the model in predicting price increases or decreases at a 5-minute level is still not particularly high, so at this point I believe there are still some deficiencies in the design structure of the model. Further adjustment of hyperparameters may be needed for model optimization in the future.

In the future, I will further adjust the structure of the model and optimize its design. In addition, combining the model signal with reinforcement learning, using the model prediction value as the state input for reinforcement learning, and combining it with PPO or DDPG models.

Using Deep learning model to generate predictive signals, while reinforcement learning model automatically generates trading strategies based on signals, which leaves to an end-to-end integrated model from raw price and volume data to trading strategies.

4 Conclusion

The comparison of cryptocurrency trading strategies we proposed revealed distinct strengths in capturing trends, exploiting market volatility, and leveraging sentiment analysis. Below is our comparison analysis.

	Momentum		Volatility			Price-Volume Divergence	Long-Short Comparison
	GC	MTS	VV	PCV	RV	PVC	LSCC
Total Return	1.45%	2.15%	1.16%	2.45%	2.74%	0.12%	1.82%
Average Daily Return	0.10%	0.15%	0.08%	0.18%	0.20%	0.87%	0.13%
Sharpe Ratio	3.79	6.71	5.10	3.41	5.39	4.34	3.98
Maximum Drawdown	-0.02	-0.01	-0.01	-0.04	-0.02	-0.002	-0.02