# Trading Concept

Pairs trading is a market-neutral trading strategy that employs a long position with a short position in a pair of highly co-moved assets.

The strategy’s profit is derived from the difference in price change between the two instruments, rather than from the direction each moves. Therefore, a profit can be realized if the long position goes up more than the short, or the short position goes down more than the long (in a perfect situation, the long position rises and the short position falls, but that’s not a requirement for making a profit). It’s possible for pairs traders to profit during a variety of market conditions, including periods when the market goes up, down or sideways — and during periods of either low or high volatility (Investopedia).

# Trading Strategy

1. Identify the cointegrated pairs. This step should be performed periodically for getting a pair (or several pairs) that will be used in the next steps.
2. Get the price history of assets, divide the data into two groups: train data for model training and test data to test model’s performance.
3. Calculate the difference between logarithm returns of assets in the pair.
4. Generating a trading signal to enter, exit the positions and which instrument to buy or sell at a specific time.
5. Test the trading signal again the buy-and-hold strategy using train data.
6. Predict the difference between logarithm returns of assets in the pair using ARIMA-GARCH model in the test data’s period of time.
7. Test the trading signal again the buy-and-hold strategy using test data.

# Trading Implementation

## Retrieve the data from Yahoo Finance

Getting historical prices of some crypto currency ['BTC-USD', 'ETH-USD', 'EOS-USD', 'LTC-USD', 'XMR-USD', 'NEO-USD', 'ZEC-USD', 'BNB-USD', 'TRX-USD'] from Yahoo Finance platform.

* Train data: from 2018-03-01 to 2018-08-31
* Test data: from 2018-09-03 to 2018-10-31

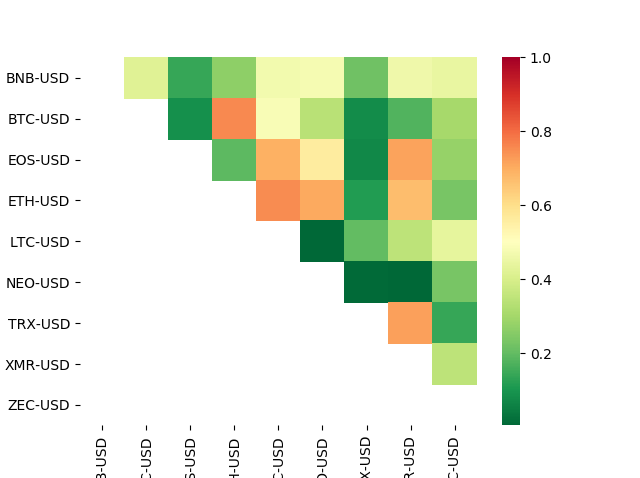
## Determine the best cointegration pair

For each currency pairs, perform the augmented Engle-Granger two-step cointegration test in order to find the best cointegration currency pair for pair trading implementation. The null hypothesis is no cointegration.

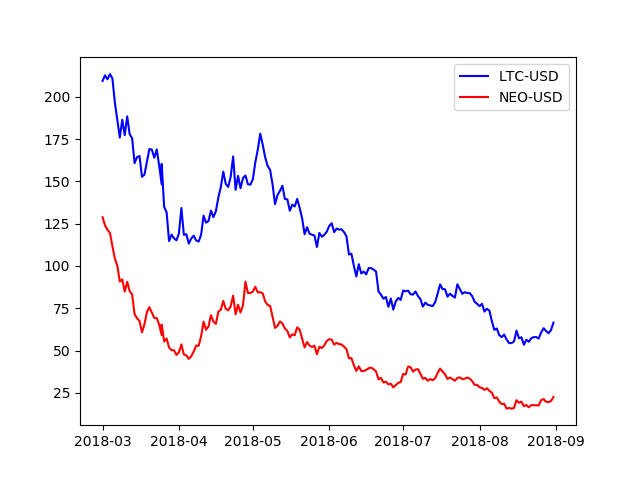
|  |  |
| --- | --- |
| **Currency Pair** | **p-value** |
| BNB-USD, BTC-USD | 0.420718222 |
| BNB-USD, EOS-USD | 0.144098571 |
| BNB-USD, ETH-USD | 0.267780112 |
| BNB-USD, LTC-USD | 0.467704922 |
| BNB-USD, NEO-USD | 0.47806333 |
| BNB-USD, TRX-USD | 0.22185604 |
| BNB-USD, XMR-USD | 0.459970843 |
| BNB-USD, ZEC-USD | 0.446940102 |
| BTC-USD, EOS-USD | 0.089401552 |
| BTC-USD, ETH-USD | 0.756574183 |
| BTC-USD, LTC-USD | 0.483869411 |
| BTC-USD, NEO-USD | 0.339223269 |
| BTC-USD, TRX-USD | 0.080514288 |
| BTC-USD, XMR-USD | 0.175564419 |
| BTC-USD, ZEC-USD | 0.307256622 |
| EOS-USD, ETH-USD | 0.192504508 |
| EOS-USD, LTC-USD | 0.696292291 |
| EOS-USD, NEO-USD | 0.563988785 |
| EOS-USD, TRX-USD | 0.070664834 |
| EOS-USD, XMR-USD | 0.718886912 |
| EOS-USD, ZEC-USD | 0.276934632 |
| ETH-USD, LTC-USD | 0.751578581 |
| ETH-USD, NEO-USD | 0.706008554 |
| ETH-USD, TRX-USD | 0.115565073 |
| ETH-USD, XMR-USD | 0.67100891 |
| ETH-USD, ZEC-USD | 0.232332592 |
| LTC-USD, NEO-USD | 0.004425966 |
| LTC-USD, TRX-USD | 0.199471903 |
| LTC-USD, XMR-USD | 0.348838689 |
| LTC-USD, ZEC-USD | 0.436279348 |
| NEO-USD, TRX-USD | 0.01008827 |
| NEO-USD, XMR-USD | 0.005272705 |
| NEO-USD, ZEC-USD | 0.232199787 |
| TRX-USD, XMR-USD | 0.722439166 |
| TRX-USD, ZEC-USD | 0.143874241 |
| XMR-USD, ZEC-USD | 0.34883363 |

Select the currency pair that has the minimum p-value, which is LTC-USD and NEO-USD.

The p-value heat map illustrates the cointegration degree of each currency pairs.



Below graph describes about the price movement of LTC-USD and NEO-USD from 2018-03-01 to 2018-08-31



Calculate the difference between the logarithm return of assets in the pairs. We choose the logarithm return over the price return because of below benefits:

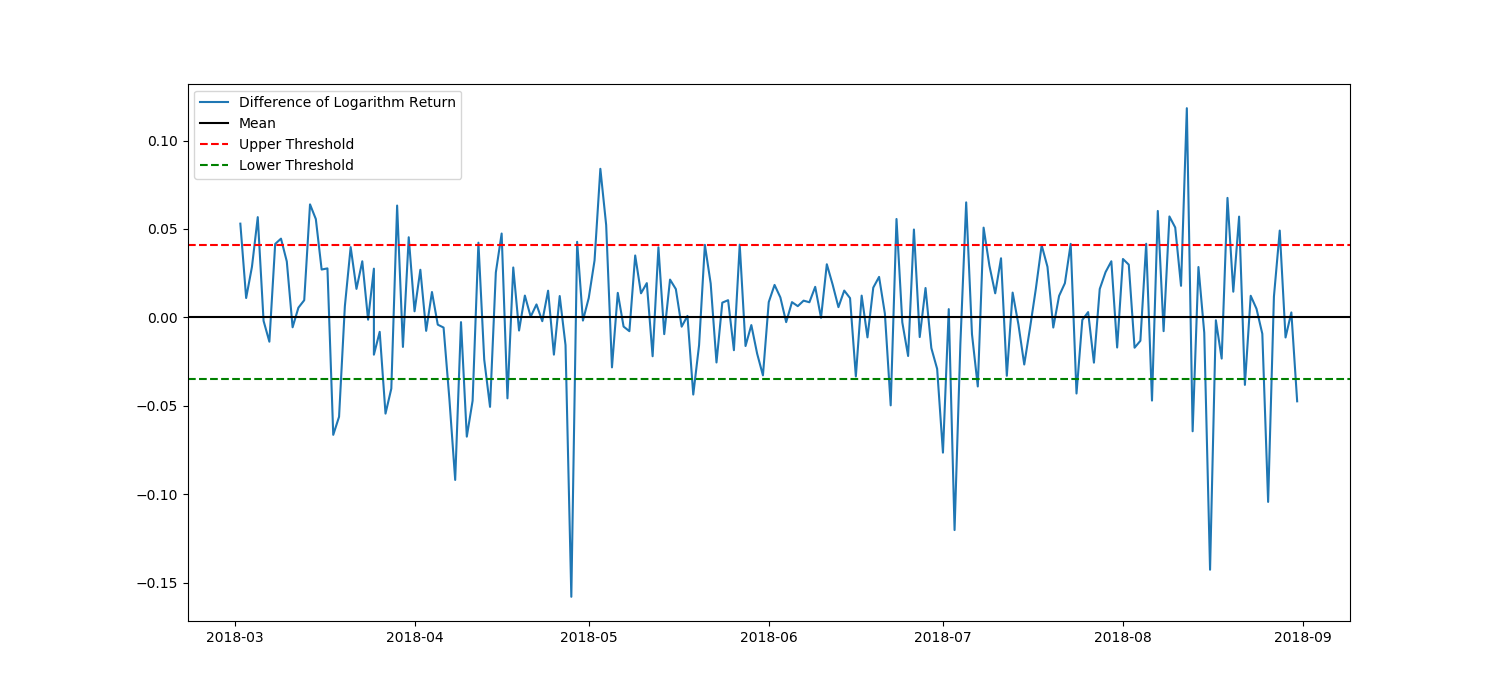
* Time additive: Note that the two-period log return is identical to the sum of the each period’s log return. To get the n-period log return, we can simply add the consecutive single period log returns. Conversely, notice the simple return is not time additive.
* Mathematically convenient: logs and exponents are easier to manipulate with calculus. Theoretical models tend to assume, unrealistically but conveniently, continuously compounded rates of return. For example, if LogReturn = LN(P1/P0), then EXP[LogReturn] = P1/P0. If f(y) = EXP[LogReturn] then the first derivative, f’(y) is quite wonderfully also EXP[LogReturn]. In short, d/dx EXP[x] = EXP[x].
* Approximately good: for short periods (e.g., daily), the log return approximates the discrete return anyway

## Determine the trading signal:

From the difference of logarithm return between two currencies, we calculate the mean and standard deviation, then set the upper threshold = mean + standard deviation and lower threshold = mean – standard deviation.

If the difference line is greater than upper threshold, the sell signal is triggered. We sell 1 LTC-USD and buy ratio number of NEO-USD. If the difference line is lower than lower threshold, the buy signal is triggered. We buy 1 LTC-USD and sell ratio number of NEO-USD.

This graph shows the differences of logarithm return line together with its mean, upper threshold and lower threshold. It gives a bird eye view regarding when the buy, sell signals are triggered.

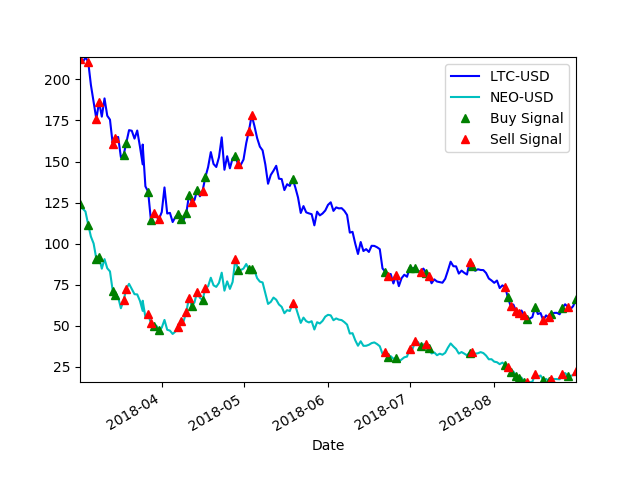


## Test trading strategy on train data

We test if our trading strategy performs better than the traditional buy-and-hold strategy or not.

* Generate the trading signal on train data
* Calculate the profit from trading strategy implementation
* Calculate the profit from buy-and-hold strategy
* Compare the profits

Following graph illustrates the trading signal that generated from our trading strategy



Our trading strategy provides the profit equals to value

The buy-and-hold strategy provides the profit equals to value

Therefore, our trading strategy is outperform the buy-and-hold strategy as we can see in the below profit chart

A close up of a map

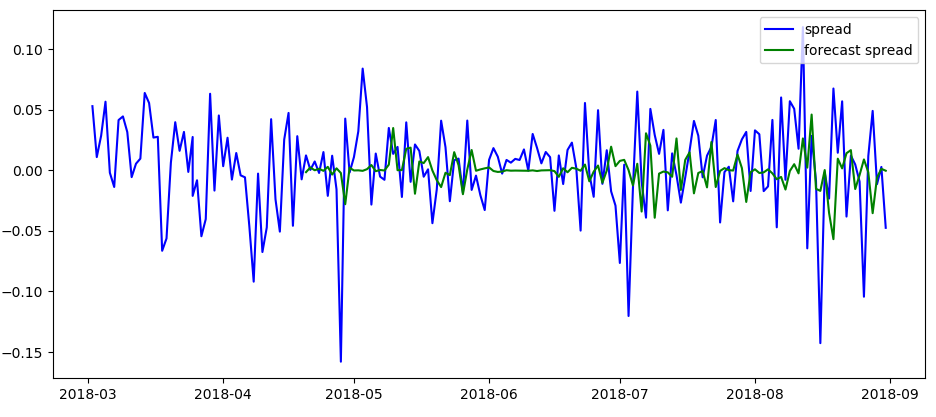
Description automatically generated

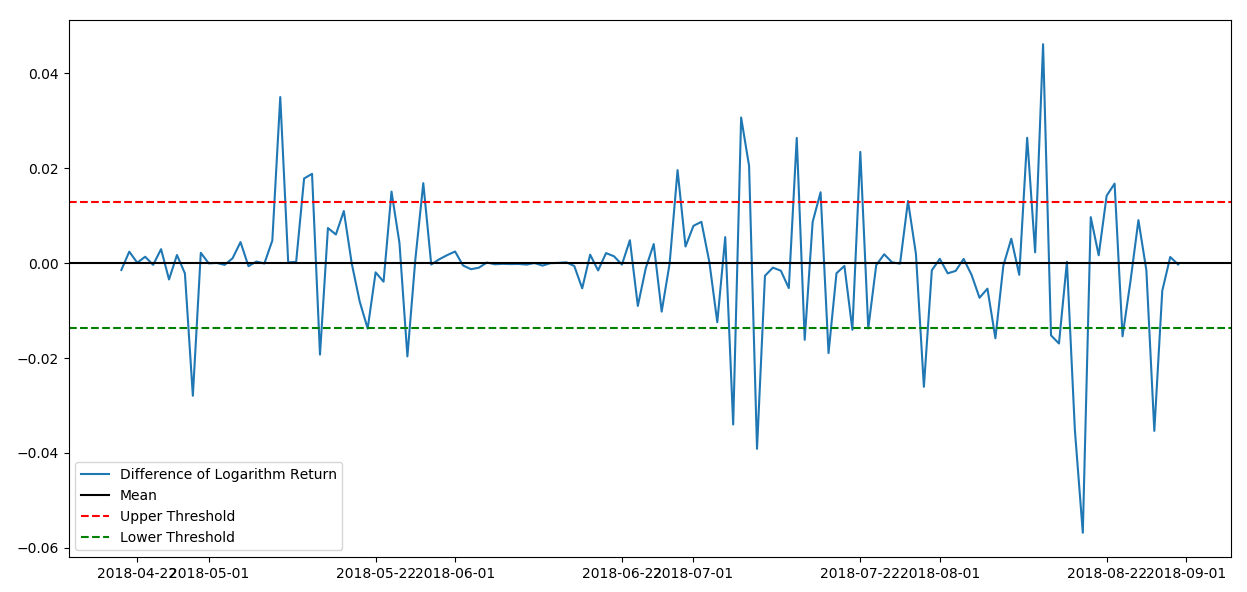
The next step is to build a model that can predict the value of differences of logarithm return, in order to generate the forecasting buy, sell signal of the next 1 day.

First, we use ARIMA model to explain the data then using the Akaike Information Critera (AIC) to select the best model.

Second, we check if the residuals has heteroskedastic behavior using Jarque\_Bera and LjungBox tests. If the residual is white noise, we use ARIMA model to predict the next day value. If the residual has heteroskedastic behavior, we need to apply the GARCH model on top of the output of ARIMA model to have a better model that can explain our time series data. We can select the best GARCH model using the Akaike Information Critera (AIC) test.

Afterward, we have to test the ARIMA-GARCH model in train data set. The following graph illustrates the difference of logarithm return which forecasted by ARIMA-GARCH model.

Next, we generate the buy, sell signal and calculate the profit from our trading strategy and buy-n-hold strategy.



Our trading strategy provides the profit equals to value

The buy-and-hold strategy provides the profit equals to value

Our trading strategy is outperform the buy-and-hold strategy as we can see in the below profit chart. Therefore, the ARIMA-GARCH model provides a good forecasting regarding the logarithm return.

## Test trading strategy on test data set

After setting up the ARIMA-GARH model and determine the trading strategy. We forecast the logarithm return and apply the trading strategy in the train data set in order to test them.

Forecasting differences of logarithm returns

Buy sell signals generated from forecasting returns

Profit of our trading signal on forecasting returns compares to the profit of trading signal compares to historical data

# Algorithm Improvement

# References