Pair Trading and Machine Learning

Overview :

The project depicts the basic form of Statistical Arbitrage, i.e Pair-Trading, in Public equity market. It relies on the assumption that two cointegrated stocks would not drift too far away from each other. We begin with selecting two stocks and apply Enger-Granger two step analysis. Once the requirement of cointegration is met, we standardize the residual and set one standard deviation(two tailed) as the threshold. Then, we compute the current standardized residuals of the selected stocks accordingly. We then look for when standardized residual excceeds the threshold, our model generates a trade signal. The catch is to go long on cheaper stock and short the expensive stock. Apparently, the interim stock spread will converge to original spread and then we will settle our position. The core idea of pair trading is **cointegration**.

Library Used for dataset:

- yahoo_fin.stock_info

Steps Overview :

- **Cointegration** : Our model uses Engle-Granger two-step method. The first step is to run a linear regression on both stocks. Next, we apply ols model and obtain the residual. Finally, we apply unit-root test to look for existence of cointegration. If the conitegration is stationary, we have found our Pair-Trading stocks.
- **Signal Generator** : This process is very straight forward. We first set a threshold limit for normalized residual. If any residual gets above or below our threshold, we go Long on bearish stock and short the bullish one, vice-versa. In this case, we only generate trading signal for one stock, the other one should be the opposite direction.
- **Plotting** : We use Matplotlib library to show some visualization. First, we plot the graph that shows residual values for our threshold limit from -1 to +1. In Figure 1 below, the yellow portion shows the time-frame when the residual is within our limit. Anytime outside the yellow portion is when our signal gets triggered for trading. The secong plot, ie Figure 2, shows our trading direction for both stocks.
- **Trade Calculation** : One thing needed to make sure for this strategy to work is that the amount for both stocks trading has to be equal. For example, we cannot buy first stock worth $1000 and sell other one worth $1200. So, in order to standardize the amount, we take a factor of 100 stocks for the first security, calculate the total amount, and then find out the number of stocks for the second security. At the end, we are left with some stocks positions in our hand. We then settle our trade positions at termination and the final amount is our profit.

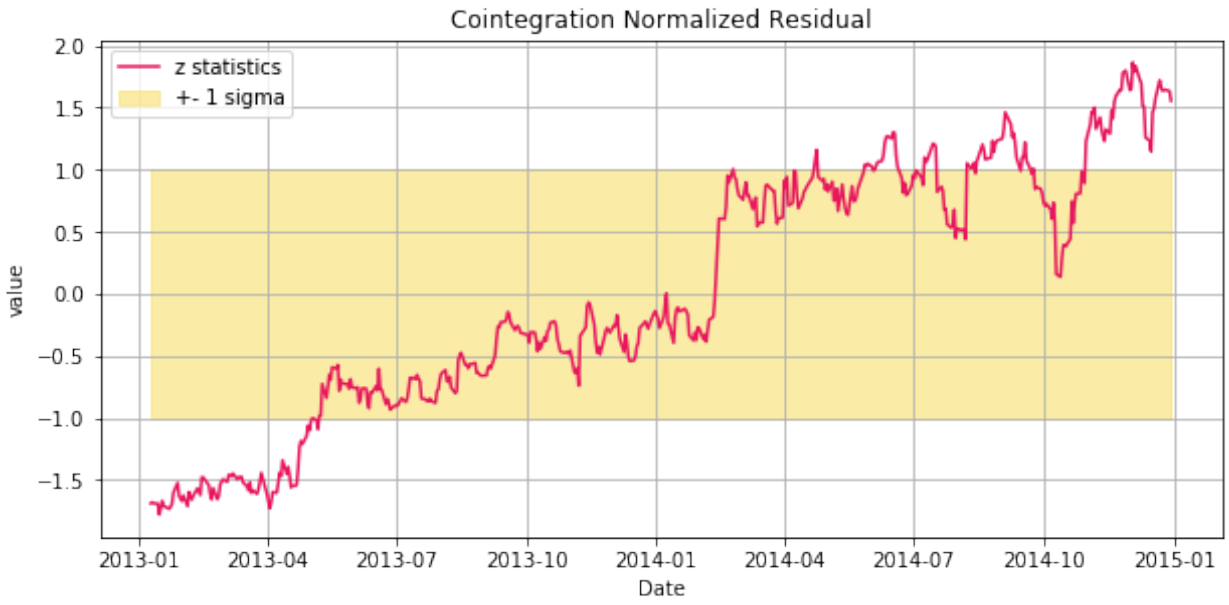

Figure 1

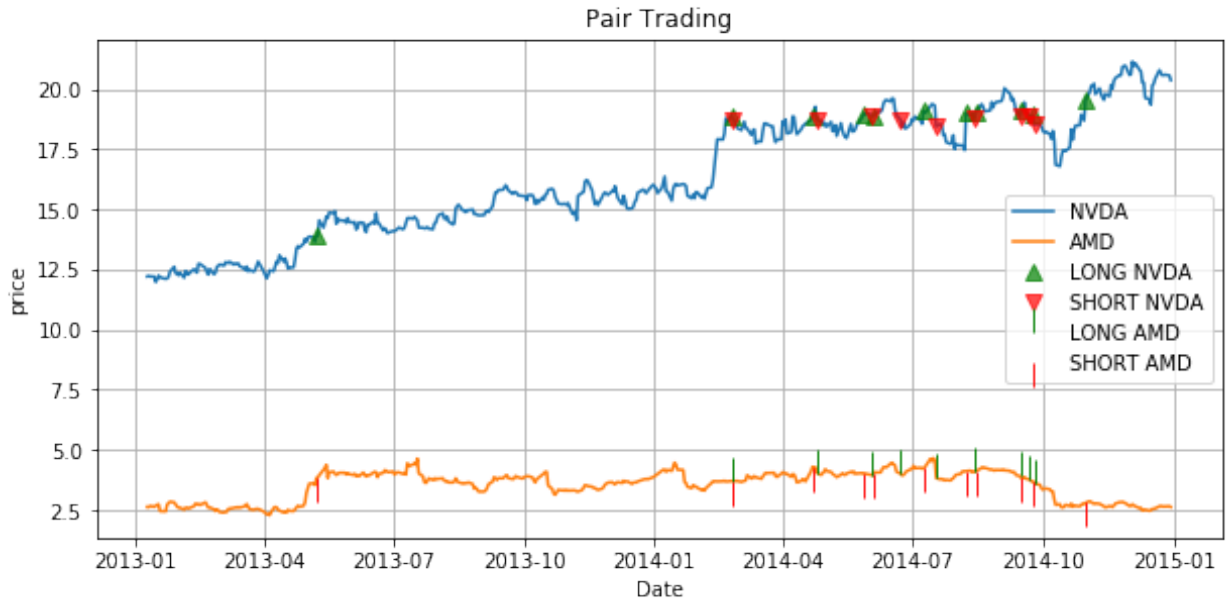

Figure 2

Maths Behind the Model:

- **Engle–Granger two-step method** : If X and Y are non-stationary and integrated of order 1, then a linear combination of them must be stationary. In other words: $\mathbf{Y_t - Beta*X_t = U_t}$**, where** $\mathbf{U_t}$ **is stationary.** A second regression is then run on the first differenced variables from the first regression, and the lagged residuals $\mathbf{U_t\text{-}1}$ is included as a regressor.

- **OLS** : **Ordinary Least Squares regression** (**OLS**) is more commonly named linear regression (simple or multiple depending on the number of explanatory variables). In the case of a model with p explanatory variables, the OLS regression model writes: $\mathbf{Y = \beta_0 + \Sigma_{j=1..p}\ \beta_j X_j + \varepsilon}$, where **Y** is the dependent variable, $\boldsymbol{\beta_0}$, is the intercept of the model, $\mathbf{X_j}$ corresponds to the j^{th} explanatory variable of the model (j= 1 to p), and e is the random error with expectation 0 and variance $\boldsymbol{\sigma^2}$. In the case where there are n observations, the estimation of the predicted value of the dependent variable Y for the i^{th} observation is given by : $\mathbf{y_i = \beta_0 + \Sigma_{j=1..p}\ \beta_j X_{ij}}$.

Results :

The following are testing criteria for our model:

- Start Date = 01/01/2013
- End Date = 12/31/2014
- Stock 1 = Nvidia
- Stock 2 = AMD

Terminal result from our model is **$14813.750720024109**, profit.

Limitations :

- Execution Risk. Slippage in price and partial order filling can hinder trading profit.
- Model Risk : Inaccurate research, flawed logic or calculations can lead to loss.
- Model doesn’t works on live data.

References :

- https://www.wikipedia.org/
- https://www.investopedia.com/
- https://stackoverflow.com/
- https://www.quantconnect.com