1. Trading Strategy

Binh Do, Robert Faff, Kais Hamza (2006) says that the spread should be defined as the difference in logarithms of the prices: $\log(P_{At}) - \log(P_{Bt})$. We ignore the margin, transaction cost in this project.

The assumption is to always trade 10,000 shares for the first stock (S1) and determine the shares for the other stock (S2) accordingly:

$$N_1P_1 + N_2P_2 = 0 \quad (1)$$

where N_1 and N_2 are the numbers of shares of S_1 and S_2, and P1 and P2 are the prices of S_1 and S_2.

Firstly, we need to compute the mean μ and standard deviation, σ, of logarithm of the ratio of the two adjusted closing stock prices in each pair from 9/1/2017 to 10/20/2017.

We will open the position to short the pair, that is, the first stock of the pair is short, and the other is long, if

$$\log\frac{P1}{P2} - \mu > k\sigma \qquad (2)$$

where k is given by the user, which has default value 1.

We will open the position to long the pair, that is, the first stock of the pair is long, and the other is short if

$$\log\frac{P1}{P2} - \mu < -k\sigma \quad (3)$$

We will close the position if the updated prices meet one of the following three formulas,

$$k1 * \sigma < \log\frac{P3}{P4} - \mu < k2 * \sigma \quad (4)$$

$$\log\frac{P3}{P4} - \mu > k3 * \sigma \quad (5)$$

$$holding\ period \leq 3 \quad (6)$$

where -k<k1<0, 0<k2<k, k3>>k, P3 and P4 are prices for S1 and S2 separately. Say k1=0.3, k2=-0.3, k3=4. Formula (4) gives a threshold for the mean reversion. Formula (5) gives a diverge tolerance. Formula (6) gives restriction that we are only allowed to hold the position no more than 3 periods.
In this way, N1 = 10,000 shares, traded at the price P1, and N2 = N1 * (P1/P2), traded at the price P2. Moreover, we close the position at P3 and P4. The profit is calculated as follows.

$$Profit = \pm N1 \times (P3 - P1) \pm N2 \times (P2 - P4) \quad (7)$$

2. Design

This project mainly focuses on three parts. Firstly, retrieve 1-year historical data and calculate the mean and standard deviation of logarithmic price ratio for each pair. Secondly, do backtesting on half a year, that is, take positions on every day and compute p/l for each pair. Thirdly, parse real time data every few minutes according to the user's preference and handle the data flow to filter the pair information, and simulate to take positions based on the historical mean and standard deviation.

The idea of this project can be visualized as Figure 2-1 shows:

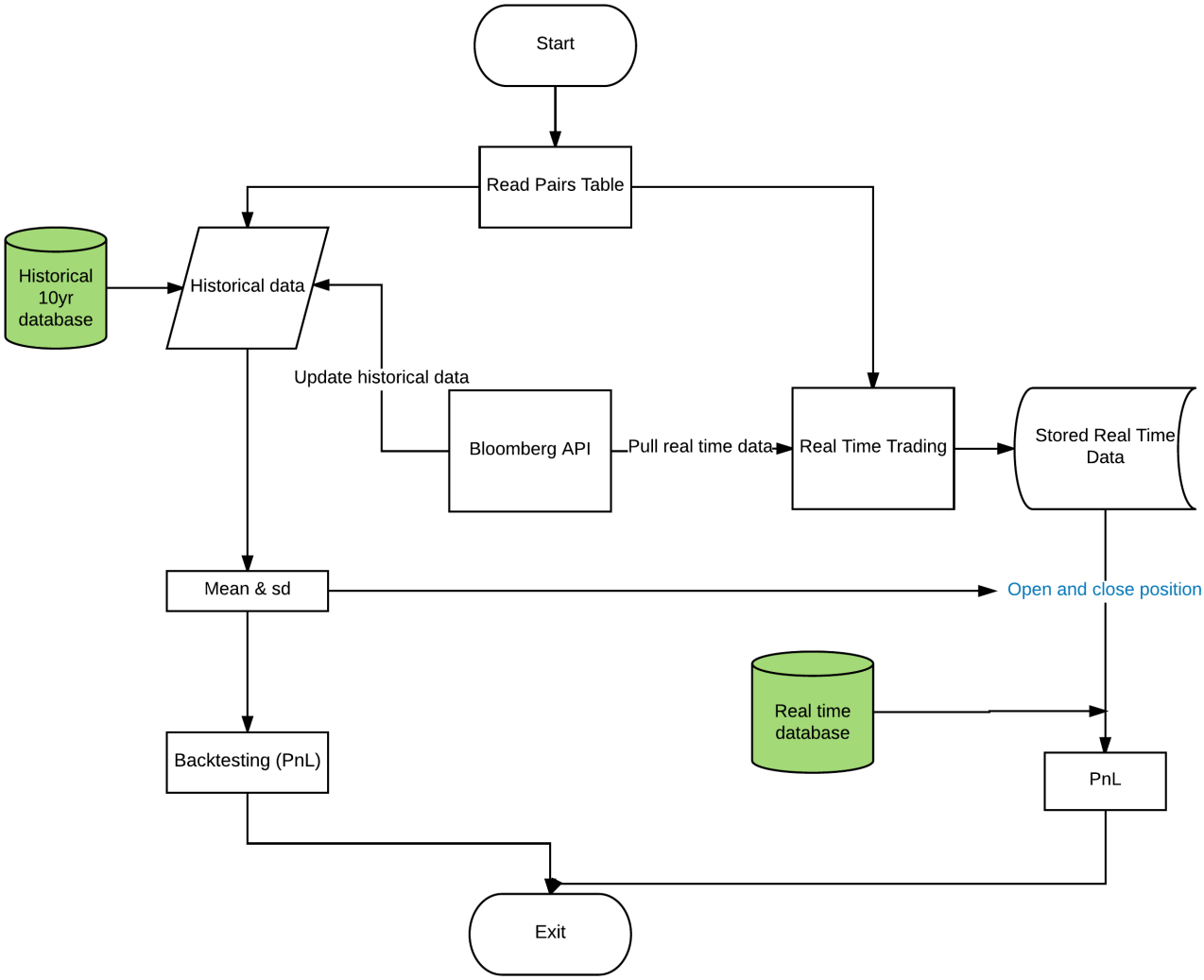

Figure 2-1 Flow chart

3. Enhancement

1) Bloomberg part

The real-time data on Bloomberg has a 15 minutes delay, so this is just a simulation to do real time trading. In this project we use the last trade price to place an order with, which is not applicable. So, we can use the bid and ask price to have more accurate positions and profits. In addition, we assume that there are always enough quantities of stocks for us to place an order in the market. Or we can change the data source Bloomberg terminal to other sources.

2) Trading strategy part

We use a very simple trading strategy here.

We can cover the mean reversion feature of each pair firstly like cointegration analysis. Then, find a more practical trading formula to open and close positions.

As for the specific trading signal formula in this project, we need to reconsider the computational method of mean and variance. If we can calculate the mean and variance based on a rolling method, the result would be more realistic. When we do real time trading, we simply choose the historical daily volatility as the real-time volatility. We know that a realistic value for the volatility leads to good estimates. So, a reasonable approach to determining the volatility to use is to look at the implied volatility of the options on the bidder stock and calculate it on a tick-by-tick basis over a given period. Or use time varying volatility based on DCC Garch Model.

3) Coding part

The coding part still needs to be polished to be more efficient and easier to use for another project.

This project is more about a simulated trading. We can write a C++ program to do real time trading, including placing an order and waiting for the order feedback. Of course, this would be much more complexed than this project.