Statistical Arbitrage Trading Strategy

Backtesting Using KDB+/Q

Kim Kuen Tang

19. April 2012

Outline

What is a trading strategy?

Mathematician William F. Donaghue: “Thorp, my advice is to buy low sell high. “

Three steps for a successful alpha model:

step	**skill**
idea	visionary
development and backtesting	quantitative and programming
successful real world implementation	entrepreneurial

This talk is about the idea, development and backtesting.

Orderbook and Trade

- An order book is the list of orders a trading venue (in particular stock exchanges) uses to record the interest of buyers and sellers.
- The trade data contains the last sale, last size and last volume of the trade.
- Use metrics of the orderbooks and trades to predict the outcomes of the returns from midprices or trades.

We are interested in the following equation

$$h_k(r_{k+1}) = f_k(\bar{o}_k, \bar{t}_k)$$

Here r_{k+1} denotes the returns, $(\bar{o}_k, \bar{t}_k)$ contains all the information of orderbooks and trades until k.

At least two ways to choose h_k and f_k

- Regression
  $r_{k+1} = \beta^{\mathrm{T}} \cdot f(\bar{o}_k, \bar{t}_k) + \epsilon_{k+1}$
- Support Vector Machine
  $sign\,(r_{k+1}) = sign\left(\beta^{\mathrm{T}} \cdot f(\bar{o}_k, \bar{t}_k) + \beta_0\right)$

This talk is not about these models. For completeness we just mentioned it here but we will not show any results.

Pairs Trading

- Identify a pair of stocks where the log prices is cointegrated
- Model the spread using an Ornstein-Uhlenbeck process.
- If observations are larger (smaller) than the predicted value we take a long (short) position and unwind it when the spread reverts.

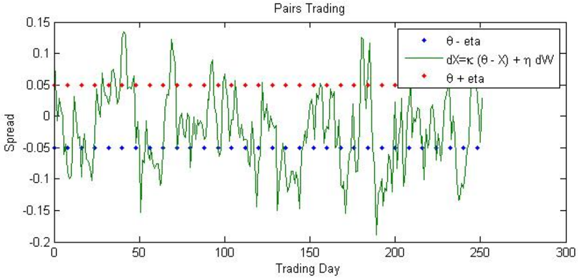

Abbildung: Example Strategy with $\theta = 0.0$

Model Specification

With $M = M_t, B = B_t, A = A_t, X = X_t, W = W_t$ and $Z = Z_t$ we describe the asset, spread and wealth dynamics by the following formulas:

- $\mathrm{d}B/B = \mu \mathrm{d}t + \sigma \mathrm{d}Z$ (price dynamic)
- $X = \ln A - \ln B$ (spread definition)
- $\mathrm{d}X = \kappa (\theta - X) \mathrm{d}t + \eta \mathrm{d}W$ (spread dynamic)
- $\mathrm{d}Z \mathrm{d}W = \rho \mathrm{d}t$ (correlation)
- $\mathrm{d}M = rM \mathrm{d}t$ (risk-free asset)

The investment behaviour is described by a process h which is progressively measureable. It models the portfolio weight for stock A and $-h$ for stock B. If V denotes the wealth dynamic of our self-financing portfolio then the dynamic is given by

$$\mathrm{d}V/V = h\mathrm{d}A/A - h\mathrm{d}B/B + r\mathrm{d}M/M$$

Observe that

$$A = B\exp(X) = f(B, X)$$

Using Ito's formula:

$$\mathrm{d}f = \dot{f}\,\mathrm{d}t + (\nabla' f)(\mathrm{d}B, \mathrm{d}X) + 0.5(\mathrm{d}B, \mathrm{d}X)(\nabla^2 f)(\mathrm{d}B, \mathrm{d}X)$$

with

$$\dot{f} = 0; f_1 = A/B; f_2 = A; f_{11} = 0; f_{12} = A/B; f_{22} = A$$

we conclude that

$$\begin{aligned} \mathrm{d}A/A - \mathrm{d}B/B = & \mathrm{d}X + \mathrm{d}B/B \cdot \mathrm{d}X + (\mathrm{d}X)^2 \\ & = \left(\kappa\left(\theta - X\right) + \eta^2/2 + \rho\sigma\eta\right)\mathrm{d}t + \eta\mathrm{d}W \end{aligned}$$

Problem Formulation

Now we seek to find the optimal position h where it is able to solve the expression below

$$\sup_{h \in \alpha(t,T)} E^{t,v,x}[U(V(T))] \text{ for an utility function } U \tag{1}$$

s. t.

$$\begin{aligned} U(x) &= x^\gamma/\gamma \text{ for a } \gamma \leq -1 \\ \mathrm{d}V/V &= h\mathrm{d}A/A - h\mathrm{d}B/B + r\mathrm{d}M/M \end{aligned}$$

Here $\alpha(t, T)$ contains all progressively and admissible processes. If we denote with $G = G(t, v, x)$ the expression in (1) then it follows with the Bellman-Principle that

$$G(t, v, x) = \sup_{h \in \alpha(t,s)} E^{t,v,x}[G(s, V(s), X(s))]$$

The Ito-integral gives the following expression

$$0 = \quad \sup_{h\in\alpha(t,s)} E^{t,v,x}\left[\int_t^s \mathrm{d}G\right]$$

Divide the right hand side by $s-t$ and let s go to t to get

$$0 = \sup_{h\in\alpha(t,T)} \text{drift}(\mathrm{d}G) = \sup_{h} h^2 \cdot \{*\} + h \cdot \{*\} + * \qquad (2)$$

The first order condition implies

$$\begin{aligned} 0 = & \quad 2 \cdot \hat{h} \cdot \{*\} + \{*\} \\ = & \quad \hat{h}\eta^2 v \cdot G_{vv} + \eta^2 \cdot G_{vx} + (-\kappa(x-\theta) + 0.5 \cdot \eta^2 \rho\sigma\eta) \cdot G_v \end{aligned}$$

Assume that $G_{vv} < 0$, it yields

$$\hat{h} = -(\eta^2 \cdot G_{v,x} + (-\kappa(x-\theta) + 0.5 \cdot \eta^2 \rho\sigma\eta) \cdot G_v)/(\eta^2 v \cdot G_{v,v}) \qquad (3)$$

Now plug back the above expression into (2) we get a differential equation which depends only on the value function G.

$$\begin{aligned}0 =&\eta^2 G_t G_{v,v} - 0.5\eta^4 G_{v,x}^2 - 0.5b^2 G_v^2 - b\eta^2 G_v G_{v,x} \\ &+ 0.5\eta^4 G_{v,v} G_{x,x} + r\eta^2 v G_v G_{v,v} - \kappa(x-\theta)\eta^2 G_x G_{v,v} \\ =&\mathfrak{L}G\end{aligned} \tag{4}$$

Here $b = -\kappa(x-\theta) + 0.5\eta^2 + \rho\sigma\eta$.
To obtain a closed form solution we will consider the following separation ansatz.

$$G(t,v,x) = f(t,x)v^\gamma$$

Here f denotes

$$f(t,x) = g(t)\exp(x\beta(t) + x^2\alpha(t))$$

Applying the Operator $\mathfrak{L}$ on the expression $g(t)\exp(x\beta(t) + x^2\alpha(t))v^\gamma$ yields

$$0 = x^2\{*\} + x\{*\} + \{*\}$$

Setting the three coefficients to zero yields the following three differential equations

$$\begin{aligned} \dot{\alpha} &= \{*\}\alpha^2 + \{*\}\alpha + \{*\} \\ 0 &= \mathfrak{B}(\alpha, \beta) \\ 0 &= \mathfrak{G}(\alpha, \beta, g) \end{aligned}$$

The first equation is a Riccati equation and $\mathfrak{B}$ and $\mathfrak{G}$ are first order linear ordinary differential equations.

With $c = \sqrt{1-\gamma}$ and $d = \exp(2\kappa(T-t)/c)$ we conclude that

$$\alpha(t) = \kappa(1-c)/(2\gamma^2) \cdot \{1 + (2c)/(1-c-(1+c)d\}$$

$$\begin{aligned}\beta(t) = \quad & 1/(2\eta^2(1-c-(1+c)d))\gamma(1-d) \cdot \\ & \{c(\eta^2 + 2\rho\sigma\eta)(1-d) - (\eta^2 + 2\rho\sigma\eta + 2\kappa\theta)\}\end{aligned}$$

$$g(t) = exp\left(-\int_t^T u(s)ds/(c^2\eta^2)\right)$$

where u satisfies a differential equation which is not of interest here.

We need to show that the expression $G = g \cdot \exp(x^2\alpha + x\beta)v^\gamma$ really satisfies the equation (3) and $G_{vv} < 0$. But this will not be shown here. Using (4) we get a closed form solution for the optimal position

$$\hat{h}(t,x) = (\beta(t) + 2x\alpha(t) - \kappa(x-\theta)/\eta^2 + \rho\sigma/\eta + 0.5)/(1-\gamma) \tag{5}$$

We will backtest this strategy with KDB+/Q.

Introduction

Choose number of path (10k) and number of points per path (5*252, 5 years of trad. days)

Tabelle: Traditional vs Vectorial Backtesting

Traditional	Vectorial
Repeat 10k times	
Sim. 2*5*252 random numbers	Sim. 10k*2*5*252 random numbers
Create the two paths	Create the 2*10k paths
Calculate the optimal positions	Calculate the 10k optimal positions
Calculate the pnl	Calculate the 10k pnl

The traditional approach is straightforward and easy to implement. Less memory is also used. But on the other hand it is also less performant. The vectorial approach will create all random numbers and store it in an offline fashion. For the vectorial approach we need a system that is able to store a lot of data in the memory.

What is KDB+/Q ?

- KDB+ is an in-memory, column-based database and Q is the query language developed by Arthur Whitney and commercialized by kx.
- Q is a terse variant of a programming language, short apl.
- The evaluation is from right to left.

We will use KDB+/Q

- to create normal distributed random numbers.
- to simulate multivariate stochastic differential equations.
- calculate pnl

Normal distributed random variables using Box Mueller Transformation

$$\sqrt{-2\ln u_1}\cdot(\cos(2\pi u_2),\sin(2\pi u_2))$$

is normal distributed for (u_1, u_2) uniform distributed between zero and one.
Code in Q.

```
wnt : raze sqrt[-2 * log n?1.0]*/:(cos; sin)@\:2 * pi * n?1.0
```

How does it work?

n?1.0	produces n uniform distr. random numbers
(cos; sin)@\: list	apply each element on left to right
*list**/: $(u; v)$	multiply each element on right to the left
raze $(u; v)$	join the two items together to form a list

Putting everything together.

$$\text{wnt : } \underbrace{\text{raze } \underbrace{\underbrace{\text{sqrt[-2} * \text{log n?1.0]}}_{\text{list of n floats}} \; \text{*/:} \; \underbrace{\text{(cos; sin)@}\backslash\text{:} \; \underbrace{2 * \text{pi} * \underbrace{\text{n?1.0}}_{\text{n random floats between 0 and 1}}}_{\text{list of n floats}}}_{\text{apply each element on left to right}}}_{\text{element-wise multiply each element on the right to the left}}}_{\text{join items, will produce a list of 2n floats}}$$

Price and Spread Process

The s.d.e. for the price process is

$$\begin{aligned} \mathrm{d}(\log S_t) &= \mathrm{d}S_t/S_t - 0.5(\mathrm{d}S_t/S_t)^2 \\ &= (\mu - 0.5\sigma^2)\mathrm{d}t + \sigma \mathrm{d}W_t \end{aligned}$$

We solve the differential equation and take equidistant time intervalls Δ for the first simulation.

$$\begin{aligned} S_{n+1} &= S_n \exp((\mu - 0.5\sigma^2)\Delta + \sigma\sqrt{\Delta}\epsilon_{n+1}) \\ &= f(S_n, \epsilon_{n+1}) \end{aligned}$$

Here ϵ_n are normal distributed random variables.

Code in Q:

$$\text{f} : \{x * \exp(h * \text{mu} - 0.5 * v * v) + v * y * \text{sqrt } h\}$$
$$\text{A} : \text{flip enlist}[st], \text{f}\backslash[\text{st} : \text{num}\# \text{ spot};\ \text{first wnt}]$$

How does it work?

{}	define a function. x is the first and y the second argument
num # spot	take num times from spot
enlist[st]	Create a list with a single element st