ภาคผนวก

คำศัพท์ Quant Trading

35 คำศัพท์สำคัญ พร้อมคำอธิบายภาษาไทย
จัดเรียงตามหมวดหมู่เพื่อใช้ค้นอ้างอิง

หมวดที่ 1 — สถิติพื้นฐาน

ค่าเฉลี่ย *Mean* μ

ผลรวมของข้อมูลทั้งหมดหารด้วยจำนวนข้อมูล บอก "จุดกึ่งกลาง" ของชุดข้อมูล ใช้เป็นจุดอ้างอิงในการวัด z-score และ spread

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i$$

→ Part 0 บทที่ 2, Part I บทที่ 7

ส่วนเบี่ยงเบนมาตรฐาน *Standard Deviation* σ

วัดว่าข้อมูลกระจายออกจากค่าเฉลี่ยมากแค่ไหน ยิ่ง σ สูง ข้อมูลยิ่งผันผวน ใช้คำนวณ z-score และ Sharpe Ratio

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2}$$

→ Part 0 บทที่ 2, Part II บทที่ 12

---

z-score z

บอกว่าค่า x ห่างจากค่าเฉลี่ยกี่หน่วยส่วนเบี่ยงเบนมาตรฐาน ใน Pair Trading ใช้ z-score ของ spread เป็นสัญญาณซื้อขาย — z < −2 หมายถึงสเปรดต่ำผิดปกติ (ซื้อ), z > +2 หมายถึงสูงผิดปกติ (ขาย)

$$z = \frac{x - \mu}{\sigma}$$

→ Part II บทที่ 14, Part IV บทที่ 22

สหสัมพันธ์ *Correlation* ρ

วัดความสัมพันธ์ระหว่างสองสิ่ง ค่าอยู่ระหว่าง −1 ถึง +1 ถ้า ρ ≈ +1 หมายถึงขึ้น-ลงพร้อมกัน ถ้า ρ ≈ −1 หมายถึงตรงข้ามกัน ถ้า ρ ≈ 0 หมายถึงไม่สัมพันธ์กัน BTC-Bybit และ BTC-Binance มี ρ สูงมาก (~0.999) จึงเหมาะทำ Pair Trading

$$\rho_{XY} = \frac{\mathrm{Cov}(X,Y)}{\sigma_X \sigma_Y}$$

→ Part 0 บทที่ 3, Part II บทที่ 13

ความแปรปรวนร่วม *Covariance*

วัดว่าตัวแปรสองตัวเปลี่ยนแปลงไปในทิศทางเดียวกันมากแค่ไหน ต่างจาก Correlation ตรงที่ไม่ได้ normalize ให้อยู่ในช่วง −1 ถึง +1 ใช้ในการคำนวณ Hedge Ratio และ Portfolio Optimization

→ Part 0 บทที่ 3, Part II บทที่ 14

การถดถอยเชิงเส้น *Linear Regression*

หาเส้นตรงที่เหมาะสมที่สุดที่อธิบายความสัมพันธ์ระหว่างตัวแปร X และ Y ใน Pair Trading ใช้ถดถอยเพื่อหา Hedge Ratio ว่าซื้อ BTC-Bybit 1 หน่วย ต้องขาย BTC-Binance กี่หน่วย

$$Y = \beta X + \alpha + \varepsilon$$

→ Part II บทที่ 13

การร่วมอินทิเกรชัน *Cointegration*

สองชุดข้อมูลแต่ละชุดอาจเดินสุ่ม (random walk) แต่ถ้าผลต่างระหว่างกันยังคงวนอยู่รอบค่าเฉลี่ย (stationary) เรียกว่าร่วมอินทิเกรต — นี่คือรากฐานทางสถิติของ Pair Trading ทดสอบด้วย ADF Test

→ Part II บทที่ 14

หมวดที่ 2 — การวัดผลกลยุทธ์

อัตราส่วนชาร์ป *Sharpe Ratio*

วัดผลตอบแทนที่ได้ต่อหน่วยความเสี่ยงที่รับ ยิ่งสูงยิ่งดี กลยุทธ์ที่ดีควรมี Sharpe > 1.0 ในระดับสากล > 2.0 ถือว่าดีมาก ตัวเลข 252 คือจำนวนวันซื้อขายต่อปีในตลาดหุ้น (หรือ 365 สำหรับ crypto)

$$\text{Sharpe} = \frac{E[R] - R_f}{\sigma_R} \times \sqrt{252}$$

R = ผลตอบแทนรายวัน, R_f = อัตราปลอดภัย (มักใช้ 0 สำหรับ crypto)

→ Part IV บทที่ 22, Part IV บทที่ 24

ดรอดาวน์สูงสุด *Maximum Drawdown (MDD)*

ขาดทุนสูงสุดจากจุดสูงสุดถึงจุดต่ำสุดในช่วงเวลาหนึ่ง วัดเป็นเปอร์เซ็นต์ บอกว่ากลยุทธ์นี้ในช่วงเลวร้ายที่สุดพอร์ตจะลดลงสูงสุดแค่ไหน กลยุทธ์ที่ดีควรมี MDD ต่ำ

$$\text{MDD} = \max_t \left(\frac{\text{peak}_t - \text{trough}_t}{\text{peak}_t} \right)$$

→ Part IV บทที่ 24

ผลตอบแทนรวมต่อปี *CAGR (Compound Annual Growth Rate)*

ผลตอบแทนเฉลี่ยต่อปีแบบทบต้น เปรียบเทียบกลยุทธ์ที่มีช่วงเวลาทดสอบต่างกันได้อย่างยุติธรรม

$$\text{CAGR} = \left(\frac{V_{\text{end}}}{V_{\text{start}}} \right)^{1/N} - 1$$

N = จำนวนปี

→ Part IV บทที่ 24

อัตราชนะ *Win Rate*

สัดส่วนการเทรดที่ได้กำไรต่อการเทรดทั้งหมด Win Rate สูงอย่างเดียวไม่พอ ต้องดูร่วมกับ Profit Factor ว่ากำไรต่อครั้งมากกว่าขาดทุนต่อครั้งหรือไม่

→ Part IV บทที่ 24

Profit Factor

ผลรวมกำไรหารด้วยผลรวมขาดทุน (เป็นบวกทั้งหมด) ถ้า > 1 แปลว่ากำไรรวมมากกว่าขาดทุนรวม ค่า > 1.5 ถือว่าดี

→ Part IV บทที่ 24

หมวดที่ 3 — แนวคิดตลาด

ส่วนต่างราคา *Spread*

ในบริบท Pair Trading: ผลต่างราคาระหว่างสองสินทรัพย์ที่คำนวณตาม Hedge Ratio เช่น Spread = BTC_Bybit − β × BTC_Binance เราซื้อขายเมื่อ spread เบี่ยงออกจากค่าเฉลี่ยมากผิดปกติ

→ Part II บทที่ 13, Part IV บทที่ 22

Bid / Ask

Bid คือราคาสูงสุดที่ผู้ซื้อยินดีจ่าย; **Ask** คือราคาต่ำสุดที่ผู้ขายยินดีรับ ส่วนต่าง Bid-Ask คือต้นทุนซ่อนเร้นของการเทรด ยิ่ง Bid-Ask ต่ำ ตลาดยิ่ง liquid

→ Part IV บทที่ 25

สลิปเปจ *Slippage*

ผลต่างระหว่างราคาที่คาดว่าจะได้กับราคาที่ได้จริงเมื่อสั่งซื้อขาย เกิดขึ้นเมื่อตลาดเคลื่อนไหวระหว่างที่คำสั่งถูกส่งและถูก fill ยิ่งคำสั่งใหญ่หรือตลาด thin ยิ่ง slippage สูง ต้องรวมใน Backtest เพื่อให้ผลลัพธ์สมจริง

→ Part IV บทที่ 25

ค่านายหน้า *Commission / Fee*

ค่าธรรมเนียมที่จ่ายให้ Exchange ต่อการเทรดหนึ่งครั้ง Bybit Taker fee ≈ 0.055% Maker fee ≈ 0.02% กลยุทธ์ที่ดีต้องคำนวณให้รวม fee แล้วยังกำไร

→ Part IV บทที่ 25, Part VI บทที่ 34

เลเวอเรจ *Leverage*

การใช้เงินกู้ขยายขนาดการเทรดให้ใหญ่กว่าทุนที่มี เช่น Leverage 10× หมายถึงใช้ทุน 1,000 USD ควบคุม Position มูลค่า 10,000 USD กำไรและขาดทุนขยายเป็น 10 เท่าตามไปด้วย

→ Part IV บทที่ 23

Long / Short

Long = ซื้อ (คาดว่าราคาจะขึ้น); **Short** = ขาย (คาดว่าราคาจะลง) ใน Pair Trading เราเปิด Long หนึ่งตัว และ Short อีกตัวพร้อมกัน ทำให้ Market Neutral — ไม่แพ้หรือชนะตามตลาดรวม แต่แพ้/ชนะตาม spread เพียงอย่างเดียว

→ Part 0 บทที่ 4, Part IV บทที่ 22

สภาพคล่อง *Liquidity*

ความสามารถซื้อขายสินทรัพย์ได้ในปริมาณมากโดยไม่กระทบราคามาก BTC มีสภาพคล่องสูงบน Bybit และ Binance ทำให้ Slippage ต่ำ เหมาะสำหรับกลยุทธ์ HFT และ Stat Arb

→ Part 0 บทที่ 4

หมวดที่ 4 — กลยุทธ์

การกลับเข้าค่าเฉลี่ย *Mean Reversion*

หลักการที่ว่าราคาหรือ spread ที่เบี่ยงออกจากค่าเฉลี่ยมักจะกลับมา รากฐานของ Pair Trading เมื่อ z-score ของ spread > +2 เราคาดว่ามันจะกลับลง จึง Short spread เมื่อ z-score < –2 เราซื้อ spread

→ Part 0 บทที่ 4, Part II บทที่ 13, Part IV บทที่ 22

โมเมนตัม *Momentum*

สิ่งที่ขึ้นมาแล้วมักขึ้นต่อ สิ่งที่ลงมาแล้วมักลงต่อ กลยุทธ์ Momentum ซื้อผู้ชนะ/ขาย Short ผู้แพ้ในช่วงที่ผ่านมา ตรงข้ามกับ Mean Reversion

→ Part III บทที่ 21, Part IV บทที่ 22

การเทรดคู่ *Pair Trading*

กลยุทธ์ Market Neutral ที่ซื้อสินทรัพย์หนึ่งและขายอีกสินทรัพย์ที่เคลื่อนไหวพร้อมกัน (cointegrated) พร้อมกัน กำไรจากการที่ spread กลับเข้าค่าเฉลี่ย ไม่ต้องทายทิศตลาด

→ Part II บทที่ 14, Part III บทที่ 21, Part IV บทที่ 22

Statistical Arbitrage *(Stat Arb)*

กลุ่มกลยุทธ์ที่ใช้หลักสถิติหา "ความผิดปกติเชิงราคา" ชั่วคราว แล้วเทรดเพื่อกำไรจากการที่ราคากลับมาถูกต้อง Pair Trading เป็น Stat Arb ประเภทหนึ่ง

→ Part 0 บทที่ 4, Part IV บทที่ 22

การตามแนวโน้ม *Trend Following*

กลยุทธ์ที่ซื้อเมื่อราคากำลังขึ้นและขาย/Short เมื่อราคากำลังลง ใช้ Moving Average, Breakout เป็นสัญญาณ ตรงข้ามกับ Mean Reversion แต่สามารถรวมกันได้ในระบบที่ซับซ้อน

→ Part IV บทที่ 22

หมวดที่ 5 — เครื่องมือสถิติ

การทดสอบ ADF *Augmented Dickey-Fuller Test*

ทดสอบว่าชุดข้อมูลเป็น *stationary* (วนรอบค่าเฉลี่ย) หรือ *random walk* ใน Pair Trading ใช้ทดสอบ spread: ถ้า p-value < 0.05 หมายถึง spread stationary — คู่นี้เหมาะทำ Pair Trading

→ Part II บทที่ 14

ครึ่งชีวิต *Half-life*

เวลาเฉลี่ยที่ spread ใช้ในการกลับมาครึ่งหนึ่งของทางจากจุดเบี่ยงสูงสุดสู่ค่าเฉลี่ย คำนวณจาก AR(1) model ใช้กำหนดความยาว Rolling Window ที่เหมาะสม

$$\text{Half-life} = -\frac{\ln 2}{\ln(1+\hat{\phi})}$$

$\hat{\varphi}$ คือ coefficient จาก AR(1): $\Delta spread_t = \varphi \cdot spread_{t-1} + \varepsilon$

→ Part II บทที่ 14, Part IV บทที่ 22

ตัวกรองคาลมาน *Kalman Filter*

อัลกอริทึมที่ประมาณค่าตัวแปร (เช่น Hedge Ratio) โดยอัปเดตทีละขั้นตอนเมื่อได้ข้อมูลใหม่ แทนที่จะใช้ Rolling Regression ที่ต้องรอให้ครบ Window ใช้ใน Pair Trading เพื่อให้ Hedge Ratio ปรับตัวตามตลาดได้เร็วกว่า

→ Part II บทที่ 17

อัตราส่วนเฮดจ์ / Beta *Hedge Ratio / β*

จำนวนหน่วยของสินทรัพย์ B ที่ต้อง Short เพื่อ offset ความเสี่ยงของการถือ Long สินทรัพย์ A หนึ่งหน่วย คำนวณจาก OLS Regression หรือ Kalman Filter ถ้า β = 0.98 หมายถึงซื้อ Bybit 1 หน่วย ขาย Binance 0.98 หน่วย

→ Part 0 บทที่ 4, Part II บทที่ 13

หน้าต่างเลื่อน *Rolling Window*

การคำนวณสถิติโดยใช้ข้อมูล N จุดล่าสุดเสมอ เช่น Rolling Mean 20 วัน หมายถึงค่าเฉลี่ยของ 20 วันที่ผ่านมา เมื่อวันใหม่เข้า วันเก่าสุดออก ทำให้สถิติปรับตัวตามสภาพตลาดปัจจุบัน

→ Part II บทที่ 12, Part IV บทที่ 22

หมวดที่ 6 — Python และเทคนิค

การเขียนโปรแกรมเชิงวัตถุ *Object-Oriented Programming (OOP)*

รูปแบบการเขียนโปรแกรมที่จัดโค้ดเป็น "วัตถุ" (Object) แต่ละวัตถุมีข้อมูล (attribute) และพฤติกรรม (method) ใน Part III ใช้ OOP สร้าง Strategy, Portfolio, และ Connector ที่สลับกันได้

→ Part III บทที่ 18–21

API *Application Programming Interface*

ส่วนต่อประสานที่ Exchange เปิดให้โปรแกรมภายนอกส่งคำสั่งและดึงข้อมูลได้ Bybit WebSocket API ส่งข้อมูลราคา real-time ทุก millisecond REST API ใช้สำหรับดึงข้อมูลประวัติและส่งคำสั่งซื้อขาย

→ Part VI บทที่ 34, Part VI บทที่ 35

อะซิงโครนัส *Async / Await*

เทคนิคเขียนโค้ดที่ให้โปรแกรมรอรับข้อมูลจาก Exchange โดยไม่ต้อง "ค้าง" รอ สามารถดูแลหลาย WebSocket พร้อมกันได้ในโปรแกรมเดียว ใช้ asyncio library ของ Python

→ Part VI บทที่ 31, Part VI บทที่ 32

DataFrame

ตารางข้อมูล 2 มิติของ pandas มี index (แถว) และ column (คอลัมน์) เป็นเครื่องมือหลักในการจัดการข้อมูลราคา เช่น OHLCV (Open, High, Low, Close, Volume) ซึ่งดาวน์โหลดจาก Exchange มาเป็นรูปแบบนี้

→ Part I บทที่ 9, Part I บทที่ 10

การทดสอบย้อนหลัง *Backtest*

การทดสอบกลยุทธ์กับข้อมูลราคาในอดีตเพื่อประเมินว่าจะทำงานอย่างไร จุดสำคัญ: ต้องระวัง Look-ahead bias (ใช้ข้อมูลอนาคตโดยไม่รู้ตัว) และ Overfitting (ปรับ parameter ให้เหมาะกับอดีตมากเกินจน ล้มเหลวในอนาคต)

→ Part IV บทที่ 22–25

Walk-Forward Optimization

วิธีทดสอบที่ซ้อม (optimize) parameter บน In-Sample data แล้วทดสอบจริงบน Out-of-Sample data ที่ไม่เคยเห็น ทำซ้ำหลายรอบแบบเลื่อนไปข้างหน้า ช่วยลด Overfitting ได้ดีกว่า Backtest ธรรมดา

→ Part IV บทที่ 24

WebSocket

โปรโตคอลการสื่อสารแบบ real-time สองทิศทาง ต่างจาก REST API ที่ต้องถามซ้ำๆ WebSocket ให้ Exchange "push" ข้อมูลใหม่มาทันทีที่มี ใช้สำหรับรับ Orderbook และราคา tick แบบ real-time

→ Part VI บทที่ 33