

บทที่ 9 การหาพอร์ตการลงทุนที่เหมาะสมที่สุด

วัตถุประสงค์การเรียนรู้

เมื่อจบบทนี้ ผู้อ่านควรสามารถ

1. แปลงปัญหาการลงทุนให้อยู่ในรูป decision variables, objective function, constraints และ data inputs
2. จำแนก convex optimization, quadratic programming และปัญหาที่ไม่เป็น convex ซึ่งพบบ่อยในการบริหารพอร์ต
3. สร้าง Minimum-Variance และ Target-Return Portfolio ภายใต้ข้อจำกัดจริง
4. ใช้เงื่อนไข Karush–Kuhn–Tucker เพื่ออธิบาย optimality และ shadow prices
5. สร้าง Maximum-Sharpe Portfolio ภายใต้ข้อจำกัด โดยไม่ใช้สูตร tangency แบบ unconstrained อย่างผิดเงื่อนไข
6. อธิบาย Maximum Diversification, Marginal Risk Contribution, Component Risk Contribution และ Risk Parity
7. ใส่ข้อจำกัดด้านน้ำหนัก กลุ่มอุตสาหกรรม leverage turnover liquidity และ transaction costs ลงใน optimizer
8. วิเคราะห์ estimation risk, covariance shrinkage, conditioning และความไม่เสถียรของน้ำหนัก
9. อธิบายหลักของ Black–Litterman และ robust optimization ในฐานะเครื่องมือจัดการ input uncertainty
10. ตรวจสอบ feasibility, solver status, KKT residuals และ numerical tolerances ก่อนนำคำตอบไปใช้
11. ออกแบบ rolling out-of-sample experiment ที่เปรียบเทียบ optimizer กับ benchmark อย่างเป็นธรรมชาติ

9.1 บทนำ: จากทฤษฎีพอร์ตสู่การตัดสินใจเชิงคำนวณ

บทที่ 6 สร้างคณิตศาสตร์ของผลตอบแทนและความเสี่ยงของพอร์ต บทที่ 7 พัฒนา minimum-variance frontier และ efficient frontier ส่วนบทที่ 8 เพิ่มสินทรัพย์ปราศจากความเสี่ยงและ tangency portfolio

บทนี้เปลี่ยนจากคำถามว่า “พอร์ตที่มีประสิทธิภาพมีสมบัติอย่างไร” ไปสู่คำถามว่า “เมื่อมีข้อจำกัดจริง เราจะคำนวณน้ำหนักที่นำไปใช้ได้อย่างไร”

ให้สัญลักษณ์พื้นฐานของปัญหาเป็น

$$\mathbf{w} = (w_1, \dots, w_n)^T, \quad \boldsymbol{\mu} = (\mu_1, \dots, \mu_n)^T, \quad \Sigma = \text{Cov}(\mathbf{R})$$

จึงมีผลตอบแทนคาดหวังและความแปรปรวนของพอร์ต

$$\mu_p = \mathbf{w}^T \boldsymbol{\mu}, \quad \sigma_p^2 = \mathbf{w}^T \Sigma \mathbf{w}$$

ในทางปฏิบัติ การหาค่า $\mathbf{w}^*$ ไม่ใช่เพียงการส่ง $\hat{\boldsymbol{\mu}}$ และ $\hat{\Sigma}$ เข้า solver แต่เป็นกระบวนการตัดสินใจที่เชื่อมโยงข้อมูล การประมาณค่า วัตถุประสงค์ ข้อจำกัด ต้นทุน และการตรวจสอบนอกตัวอย่างเข้าด้วยกัน ความผิดพลาดจึงเกิดได้อย่างน้อยสามระดับ ได้แก่ *input risk* จากค่าประมาณที่คลาดเคลื่อน *model risk* จาก objective หรือ constraints ที่ไม่ตรงกับงานจริง และ *implementation risk* จากต้นทุน สภาพคล่อง และข้อจำกัดในการส่งคำสั่งซื้อขาย

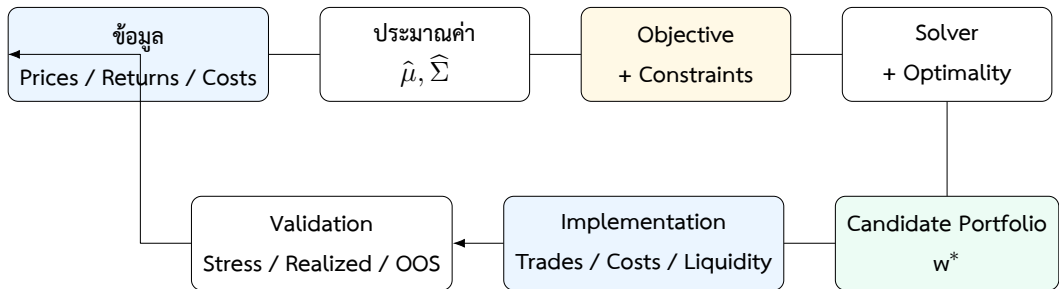
ทุกพจน์ใน objective ต้องใช้หน่วยและช่วงเวลาเดียวกัน ตัวอย่างเช่น ถ้า $\boldsymbol{\mu}$ เป็นผลตอบแทนคาดหวังรายปี Σ ก็ควรเป็น covariance รายปี และ transaction-cost term ที่นำมาหักต้องถูกแปลงให้อยู่ในหน่วยผลตอบแทนที่สอดคล้องกัน

บทนิยาม 9.1

ปัญหาการหาพอร์ตที่เหมาะสมที่สุดคือการเลือกเวกเตอร์น้ำหนัก $\mathbf{w}$ เพื่อให้ objective function ดีที่สุดภายใต้ feasible set $\mathcal{F}$

$$\min_{\mathbf{w} \in \mathcal{F}} f(\mathbf{w}) \quad \text{หรือ} \quad \max_{\mathbf{w} \in \mathcal{F}} g(\mathbf{w})$$

คำว่า “เหมาะสมที่สุด” จึงมีความหมายเฉพาะเมื่อ objective, inputs และ constraints ถูกระบุอย่างครบถ้วน



รูปที่ 9.1: วงจร Portfolio Optimization ตั้งแต่ข้อมูล การประมาณค่า การสร้างพอร์ต การนำไปใช้ และการตรวจสอบนอกตัวอย่าง

ประเด็นที่ต้องระวัง

Solver สามารถหาคำตอบของปัญหาที่กำหนดได้อย่างแม่นยำ แต่ไม่สามารถแก้ปัญหา input ที่ไม่เหมาะสม objective ที่ไม่ตรงกับเป้าหมาย หรือ constraints ที่ไม่สะท้อนการซื้อขายจริงได้

9.2 โครงสร้างของ Portfolio Optimization Problem

ปัญหาหนึ่งควรถูกแยกเป็นสี่องค์ประกอบ

1. **Decision variables** เช่น น้ำหนัก w_i หรือ trade vector Δw
2. **Objective function** เช่น variance, utility, tracking error, transaction cost หรือ risk concentration
3. **Constraints** เช่น budget, long-only, position limits, sector limits, leverage, turnover และ liquidity
4. **Inputs** เช่น $\hat{\mu}, \hat{\Sigma}$, benchmark weights, current holdings และ cost estimates

9.2.1 Convexity และเหตุใดจึงสำคัญ

บทนิยาม 9.2

เซต $\mathcal{F}$ เป็น convex set ถ้าสำหรับ $\mathbf{w}^{(1)}, \mathbf{w}^{(2)} \in \mathcal{F}$ และ $\theta \in [0, 1]$

$$\theta \mathbf{w}^{(1)} + (1 - \theta) \mathbf{w}^{(2)} \in \mathcal{F}$$

ฟังก์ชัน f เป็น convex ถ้า

$$f(\theta \mathbf{w}^{(1)} + (1 - \theta) \mathbf{w}^{(2)}) \leq \theta f(\mathbf{w}^{(1)}) + (1 - \theta) f(\mathbf{w}^{(2)})$$

ถ้า objective เป็น convex และ feasible set เป็น convex จุดต่ำสุดเฉพาะที่ใด ๆ ที่มีอยู่จะเป็นจุดต่ำสุดทั่วทั้งเซต นี่คือเหตุผลสำคัญที่ปัญหา minimum variance พร้อมข้อจำกัดเชิงเส้นได้รับความนิยม เพราะสามารถตรวจ optimality ได้อย่างเป็นระบบและไม่ต้องกังวลกับ local minima หลายจุดแบบปัญหา non-convex

ทฤษฎีบท 9.1

ถ้า $\Sigma \succeq 0$ แล้ว

$$f(\mathbf{w}) = \mathbf{w}^T \Sigma \mathbf{w}$$

เป็น convex function

พิสูจน์. เพราะ Σ symmetric

$$\nabla f(\mathbf{w}) = 2\Sigma\mathbf{w}$$

และ

$$\nabla^2 f(\mathbf{w}) = 2\Sigma$$

เมื่อ $\Sigma \succeq 0$ Hessian เป็น positive semidefinite จึงได้ว่า f เป็น convex □

ถ้า $\Sigma \succ 0$ ฟังก์ชันเป็น strictly convex ดังนั้นถ้า optimum มีอยู่บน convex feasible set คำตอบจะมี

เอกลักษณ์ สำหรับ long-only fully invested portfolio feasible set เป็น simplex ที่ปิดและมีขอบเขต จึงรับประกันการมี minimum เมื่อ objective ต่อเนื่อง ส่วนปัญหาที่มี leverage หรือ short selling แบบไม่มีขอบเขต ต้องตรวจ existence และ boundedness แยกต่างหาก

9.3 Minimum-Variance Portfolio ภายใต้ข้อจำกัด

ปัญหา long-only minimum variance คือ

Long-Only Minimum Variance

$$\begin{array}{ll} \min_{\mathbf{w}} & \frac{1}{2} \mathbf{w}^T \Sigma \mathbf{w} \\ \text{s.t.} & \mathbf{1}^T \mathbf{w} = 1 \\ & \mathbf{w} \geq 0 \end{array}$$

ตัวคูณ $1/2$ ไม่เปลี่ยนคำตอบ แต่ทำให้ gradient เท่ากับ $\Sigma \mathbf{w}$

ถ้าไม่มี long-only constraint และ $\Sigma \succ 0$ จะกลับไปสู่คำตอบ GMV ในบทที่ 7

$$\mathbf{w}_{\text{GMV}} = \frac{\Sigma^{-1} \mathbf{1}}{\mathbf{1}^T \Sigma^{-1} \mathbf{1}}$$

แต่เมื่อมี inequalities สูตรปิดรูปร่างนี้อาจไม่ใช่ feasible solution หากน้ำหนักบางตัวติดลบหรือเกิน position limits ต้องแก้ constrained problem ใหม่ จุดสำคัญคือ “ข้อจำกัด” ไม่ได้เป็นขั้นตอนแก้ไขน้ำหนักหลัง optimization แต่เป็นส่วนหนึ่งของนิยามปัญหาดั้งแต่ต้น การตัดน้ำหนักติดลบให้เป็นศูนย์แล้ว normalize ใหม่โดยไม่ re-optimize โดยทั่วไปไม่ให้คำตอบ minimum variance ภายใต้ข้อจำกัดใหม่

9.4 KKT Conditions และความหมายของ Active Constraints

สำหรับ long-only minimum variance กำหนด Lagrangian

$$\mathcal{L} = \frac{1}{2} \mathbf{w}^T \Sigma \mathbf{w} - \lambda (\mathbf{1}^T \mathbf{w} - 1) - \nu^T \mathbf{w}$$

โดย

$$\nu \geq 0$$

เงื่อนไข KKT คือ

$$\begin{aligned} \Sigma \mathbf{w} - \lambda \mathbf{1} - \nu &= 0, & \mathbf{1}^T \mathbf{w} &= 1, \\ \mathbf{w} &\geq 0, & \nu &\geq 0, & \nu_i w_i &= 0 \quad (i = 1, \dots, n) \end{aligned}$$

โดยสมการแรกคือ stationarity สมการที่สองคือ primal feasibility เงื่อนไข $\nu \geq 0$ คือ dual feasibility และ $\nu_i w_i = 0$ คือ complementary slackness

บทนิยาม 9.3

ถ้า $w_i > 0$ complementary slackness บังคับให้

$$\nu_i = 0$$

ถ้า $w_i = 0$ constraint ของสินทรัพย์นั้นอาจ active และ ν_i สามารถเป็นบวกได้ การเปลี่ยน active set เมื่อ parameters หรือ target return เปลี่ยนเป็นสาเหตุหนึ่งที่ทำให้ constrained frontier มี corner portfolios

ตัวคูณ Lagrange และ KKT multipliers ยังมีความหมายเชิงเศรษฐศาสตร์ในฐานะ *shadow prices* กล่าวคือ บอกการเปลี่ยนแปลงของค่าที่เหมาะสมที่สุดโดยประมาณเมื่อผ่อนหรือทำให้ constraint เข้มข้นเล็กน้อย อย่างไรก็ตาม เครื่องหมายของ shadow price ขึ้นกับรูปที่ใช้เขียน constraint และการตีความเชิงอนุพันธ์ใช้ได้ภายในช่วงที่ active set ไม่เปลี่ยน

ข้อสังเกต

ภายใต้ convexity และ constraint qualification ที่เหมาะสม KKT conditions เป็นเงื่อนไขที่ใช้ระบุ global optimum ได้ ไม่ใช่เพียงเงื่อนไขอันดับหนึ่งของ local optimum

9.5 Target-Return Portfolio และ Constrained Efficient Frontier

ให้ target return เท่ากับ μ_0

Target-Return Minimum Variance

$$\begin{array}{ll} \min_{\mathbf{w}} & \frac{1}{2} \mathbf{w}^T \Sigma \mathbf{w} \\ \text{s.t.} & \mathbf{1}^T \mathbf{w} = 1 \\ & \mu^T \mathbf{w} \geq \mu_0 \\ & \ell_i \leq w_i \leq u_i \end{array}$$

รูปแบบ $\mu^T \mathbf{w} \geq \mu_0$ ควรตีความว่าเป็น *minimum required return* ถ้า μ_0 สูงกว่าผลตอบแทนของ constrained GMV portfolio และปัญหาเป็น convex ตามเงื่อนไขมาตรฐาน ข้อจำกัดผลตอบแทนจะ bind ที่คำตอบ จึงได้ $\mu^T \mathbf{w} = \mu_0$ แต่ถ้า μ_0 ต่ำกว่าผลตอบแทนของ GMV constraint นี้จะไม่ active และ optimizer จะคืน GMV portfolio เดิม

ดังนั้นการสร้าง constrained efficient frontier สามารถทำได้โดยแก้ปัญหาซ้ำบนกริดของ μ_0 ที่เริ่มจากผลตอบแทนของ constrained GMV ขึ้นไป หรือใช้ equality $\mu^T \mathbf{w} = \mu_0$ เมื่อประสงค์ให้พอร์ตมี expected return เท่ากับ target โดยตรง

ทฤษฎีบท 9.2

ถ้า

$$\mathcal{F}_2 \subseteq \mathcal{F}_1$$

แล้วสำหรับ target return เดียวกัน

$$\min_{\mathbf{w} \in \mathcal{F}_2} \mathbf{w}^T \Sigma \mathbf{w} \geq \min_{\mathbf{w} \in \mathcal{F}_1} \mathbf{w}^T \Sigma \mathbf{w}$$

เมื่อทั้งสองปัญหามี feasible solution

ผลนี้อธิบายว่า constraints ทำให้ in-sample opportunity set แคบลง แต่ไม่ได้หมายความว่า constraints ไม่มีประโยชน์ เพราะ constraints สามารถลด concentration, leverage, turnover และ estimation-error amplification

9.5.1 Feasibility ก่อน Optimization

สำหรับ long-only fully invested portfolio

$$\mathbf{w} \geq 0 \quad \mathbf{1}^T \mathbf{w} = 1$$

จะมี

$$\min_i \mu_i \leq \mu^T \mathbf{w} \leq \max_i \mu_i$$

ดังนั้น target return ที่สูงกว่า $\max_i \mu_i$ เป็น infeasible ทันที

เมื่อมี upper bounds หรือ group constraints feasible return interval อาจแคบกว่านี้ วิธีที่ทั่วไปกว่าและตรวจสอบได้คือคำนวณ

$$\mu_{\min}^{\mathcal{F}} = \min_{\mathbf{w} \in \mathcal{F}} \mu^T \mathbf{w}, \quad \mu_{\max}^{\mathcal{F}} = \max_{\mathbf{w} \in \mathcal{F}} \mu^T \mathbf{w}$$

ก่อนสร้าง target grid ถ้า $\mathcal{F}$ นิยามด้วยข้อจำกัดเชิงเส้น ปัญหาสองข้อนี้เป็น linear programs และช่วยแยก “target ที่เป็นไปไม่ได้” ออกจาก “solver ที่แก้ปัญหาไม่สำเร็จ” ได้อย่างชัดเจน

ประเด็นที่ต้องระวัง

สถานะ “infeasible” ไม่ได้หมายความว่า solver ล้มเหลวเสมอไป แต่อาจหมายถึงข้อกำหนดของผู้ใช้ขัดกันจริง เช่น target return สูงเกินไปพร้อมกับ position caps ที่เข้มเกินไป

9.6 Maximum-Sharpe Portfolio ภายใต้ข้อจำกัด

เมื่อมี risk-free rate R_f และ $\mathbf{w}$ แทนน้ำหนักของ risky assets ที่ fully invested นิยาม Sharpe ratio สำหรับพอร์ตที่มี $\sigma_p > 0$ เป็น

$$S(\mathbf{w}) = \frac{\mu^T \mathbf{w} - R_f}{\sqrt{\mathbf{w}^T \Sigma \mathbf{w}}}$$

ตัวเลขต้องใช้ผลตอบแทนและ R_f ที่มี horizon และ convention เดียวกัน เช่น nominal ต่อปีทั้งคู่

ในกรณี unconstrained บทที่ 8 ให้ tangency direction

$$w_T \propto \Sigma^{-1}(\mu - R_f \mathbf{1})$$

แต่เมื่อมี constraints ต้องแก้

$$\max_{w \in \mathcal{F}} S(w)$$

ใหม่

ประเด็นที่ต้องระวัง

Sharpe ratio เป็น fractional objective จึงไม่ใช่ quadratic objective โดยตรง การใส่ Sharpe ratio เข้า quadratic-programming solver โดยไม่ reformulate เป็นข้อผิดพลาดเชิงแบบจำลอง

แนวทางเชิงคำนวณที่ใช้ได้ขึ้นกับข้อจำกัด เช่น nonlinear optimization หรือการแก้ target excess-return minimum-variance problems หลายระดับแล้วเลือกจุดที่มี Sharpe ratio สูงที่สุด วิธีหลังมีข้อดีเชิงการสอน เพราะสามารถตรวจ feasibility และ variance ของแต่ละจุดบน constrained frontier ได้โดยตรง

ถ้า expected excess return ของทุกพอร์ต feasible ไม่เป็นบวก ค่า maximum Sharpe อาจเป็นศูนย์หรือติดลบ และแนวคิด “เส้นสัมผัสจาก R_f ” ต้องตีความด้วยความระมัดระวัง ไม่ควรบังคับใช้สูตร tangency โดยสมมติว่าพอร์ตที่มี Sharpe ratio บวกต้องมีอยู่เสมอ

9.7 Maximum Diversification Portfolio

ให้

$$\sigma = (\sigma_1, \dots, \sigma_n)^T$$

Diversification Ratio นิยามเป็น

$$DR(w) = \frac{w^T \sigma}{\sqrt{w^T \Sigma w}}$$

Maximum Diversification Portfolio แก้

$$\max_{\mathbf{w} \in \mathcal{F}} DR(\mathbf{w})$$

ข้อสังเกต

สำหรับ long-only fully invested portfolio ตัวเลข $\mathbf{w}^T \boldsymbol{\sigma}$ คือ weighted average ของ standalone volatilities ส่วนตัวส่วนคือ volatility หลังรวม covariance จึงวัดประโยชน์จากโครงสร้างความสัมพันธ์มากกว่าจำนวนสินทรัพย์เพียงอย่างเดียว ถ้าสินทรัพย์เคลื่อนไหวสัมพันธ์กัน สมบูรณ์ในทิศทางเดียว Diversification Ratio จะไม่สร้างประโยชน์จาก correlation แต่เมื่อ correlation ต่ำลง อัตราส่วนนี้สามารถสูงกว่า 1 ได้

ภายใต้ long-only และ fully invested สามารถใช้ numerical optimization หรือ reformulation ที่เหมาะสมเพื่อแก้ปัญหา โดยต้องตรวจ normalization และ constraints ของ formulation ที่ใช้ หากอนุญาต short selling ตัวเลขอาจไม่ใช่ weighted average ที่มีความหมายเดิม จึงต้องระบุ domain และการตีความของ Diversification Ratio ให้ชัดเจน

งานเชิงประจักษ์ด้าน risk models และ diversification แสดงการใช้ minimum variance, maximum diversification และ risk parity กับ covariance estimators หลายรูปแบบ พร้อม upper bounds เพื่อควบคุม concentration

9.8 Risk Contribution และ Risk Parity

portfolio volatility คือ

$$\sigma_p = \sqrt{\mathbf{w}^T \boldsymbol{\Sigma} \mathbf{w}}$$

สำหรับ $\sigma_p > 0$ Marginal Risk Contribution ของสินทรัพย์ i คือ

$$MRC_i = \frac{\partial \sigma_p}{\partial w_i} = \frac{(\boldsymbol{\Sigma} \mathbf{w})_i}{\sigma_p}$$

Component Risk Contribution คือ

$$RC_i = w_i MRC_i = \frac{w_i (\boldsymbol{\Sigma} \mathbf{w})_i}{\sigma_p}$$

จาก Euler's theorem

$$\sum_{i=1}^n RC_i = \sigma_p$$

กำหนด percentage risk contribution

$$PRC_i = \frac{RC_i}{\sigma_p}$$

จึงมี

$$\sum_{i=1}^n PRC_i = 1$$

บทนิยาม 9.4

Equal Risk Contribution Portfolio หรือ risk-parity portfolio ในรูปพื้นฐานต้องการ

$$RC_i = \frac{\sigma_p}{n}$$

สำหรับสินทรัพย์ที่อยู่ในพอร์ต

กรณีมี risk budgets $b_i > 0$ และ $\sum_i b_i = 1$ จะกำหนดเป้าหมาย

$$RC_i = b_i \sigma_p$$

หรือในรูปที่ไม่ต้องหารด้วย σ_p

$$w_i(\Sigma w)_i = b_i w^T \Sigma w$$

แม้ $\sum_i RC_i = \sigma_p$ เสมอตาม Euler decomposition แต่ RC_i แต่ละตัวไม่จำเป็นต้องเป็นบวก ในพอร์ตที่อนุญาต short selling หรือมีสินทรัพย์ที่ทำหน้าที่ hedge อาจเกิด $MRC_i < 0$ และ $RC_i < 0$ ได้ ดังนั้นการสร้าง risk-budgeting portfolio แบบมาตรฐานมักกำหนด long-only และ positive risk budgets เพื่อให้การตีความ “ส่วนแบ่งความเสี่ยง” ตรงไปตรงมา

ประเด็นที่ต้องระวัง

Equal weight ไม่เท่ากับ equal risk contribution โดยทั่วไป และ inverse-volatility weights จะเท่ากับ risk parity เฉพาะภายใต้โครงสร้าง covariance บางกรณี ไม่ใช่เอกลักษณ์ทั่วไป

9.9 ข้อจำกัดที่พบในการบริหารพอร์ตจริง

9.9.1 Position Bounds

$$\ell_i \leq w_i \leq u_i$$

upper bounds ช่วยลด concentration และลดความไวของ optimizer ต่อ input errors

9.9.2 Group และ Sector Limits

ถ้า $\mathcal{G}_j$ เป็นกลุ่มสินทรัพย์

$$L_j \leq \sum_{i \in \mathcal{G}_j} w_i \leq U_j$$

9.9.3 Gross Exposure และ Leverage

สำหรับ long-short portfolio

$$\|\mathbf{w}\|_1 = \sum_i |w_i| \leq G_{\max}$$

ใน fully invested portfolio

$$\mathbf{1}^T \mathbf{w} = 1$$

net exposure กับ gross exposure จึงเป็นคนละแนวคิด ตัวอย่างเช่นพอร์ตที่มี long 150% และ short 50% มี net exposure 100% แต่ gross exposure 200% การควบคุมเพียง $\mathbf{1}^T \mathbf{w} = 1$ จึงไม่เพียงพอสำหรับพอร์ต

long-short

9.9.4 Tracking Error Constraint

ให้ benchmark weights เป็น $\mathbf{w}_b$

$$TE(\mathbf{w}) = \sqrt{(\mathbf{w} - \mathbf{w}_b)^T \Sigma (\mathbf{w} - \mathbf{w}_b)}$$

สามารถกำหนด

$$TE(\mathbf{w}) \leq TE_{\max}$$

โดย Σ ควรเป็น covariance ที่สอดคล้องกับ active returns และ horizon ของ mandate ถ้าพอร์ตและ benchmark fully invested ทั้งคู่ จะมี $\mathbf{1}^T (\mathbf{w} - \mathbf{w}_b) = 0$ ทำให้ active risk มาจากการเบี่ยงน้ำหนักจาก benchmark ไม่ใช่จากการเพิ่มหรือลดเงินสดโดยไม่ตั้งใจ

9.9.5 Turnover Constraint

ให้ $\mathbf{w}^{\text{old}}$ เป็นน้ำหนักก่อน rebalance

$$TO(\mathbf{w}) = \sum_i |w_i - w_i^{\text{old}}|$$

และกำหนด

$$TO(\mathbf{w}) \leq \tau$$

ข้อสังเกต

บางสถาบันนิยม one-way turnover เป็นครึ่งหนึ่งของผลรวม absolute trades

$$TO_{\text{oneway}} = \frac{1}{2} \sum_i |w_i - w_i^{\text{old}}|$$

จึงต้องระบุ convention ก่อนเปรียบเทียบตัวเลข turnover โดยเฉพาะเมื่อมี external cash flow เพราะ

เงื่อนไข “มูลค่าซื้อเท่ากับมูลค่าขาย” อาจไม่เป็นจริงในเวลานั้น

9.10 Transaction Costs และการตัดสินใจจาก Holdings เดิม

เมื่อมี current portfolio การตัดสินใจที่แท้จริงคือ trade vector

$$\Delta \mathbf{w} = \mathbf{w} - \mathbf{w}^{\text{old}}$$

linear transaction-cost approximation คือ

$$C_{\text{lin}}(\Delta \mathbf{w}) = \sum_i c_i |\Delta w_i|$$

และแบบจำลองที่เพิ่ม market impact อาจใช้

$$C(\Delta \mathbf{w}) = \sum_i [c_i |\Delta w_i| + q_i (\Delta w_i)^2]$$

โดย $c_i \geq 0$ และ $q_i \geq 0$ พจน์ $c_i |\Delta w_i|$ สามารถใช้แทนค่าธรรมเนียมและ bid-ask spread โดยประมาณ ส่วนพจน์กำลังสองใช้เป็น local approximation ของ market impact ต้นทุนจริงอาจไม่สมมาตรระหว่างซื้อกับขายและอาจไม่เป็น quadratic จึงควรประมาณค่าจากข้อมูลการซื้อขายที่เหมาะสมกับขนาดคำสั่ง

ปัญหา utility หลังต้นทุนสามารถเขียนเป็น

Return-Risk-Cost Optimization

$$\max_{\mathbf{w} \in \mathcal{F}} \left\{ \mu^T \mathbf{w} - \frac{\lambda}{2} \mathbf{w}^T \Sigma \mathbf{w} - C(\mathbf{w} - \mathbf{w}^{\text{old}}) \right\}$$

เมื่อ $\lambda > 0$ เป็น risk-aversion parameter และทุกพจน์ในวงเล็บถูกแปลงให้อยู่ในหน่วยผลตอบแทนของ horizon เดียวกัน หาก $\Sigma \succeq 0$ และ $C(\cdot)$ เป็น convex ฟังก์ชันที่ maximize ข้างต้นเป็น concave objective จึงยังอยู่ในกรอบ convex optimization เมื่อ feasible set เป็น convex

ข้อสังเกต

การมี transaction-cost penalty ทำให้ optimal target weights ขึ้นกับ current holdings ด้วย ดังนั้น นักลงทุนสองรายที่มี expectations เหมือนกันอาจมี optimal trades ต่างกันเพราะเริ่มจาก holdings คนละชุด

9.11 Liquidity Constraints

ถ้ามูลค่าพอร์ตเป็น V และสินทรัพย์ i มี Average Daily Value Traded เท่ากับ ADV_i อาจกำหนด trade limit

$$V|\Delta w_i| \leq \kappa_i ADV_i$$

เมื่อ κ_i เป็นสัดส่วน participation ที่ยอมรับได้

หรือเขียน

$$|\Delta w_i| \leq \frac{\kappa_i ADV_i}{V}$$

และสามารถตีความ “จำนวนวันโดยประมาณที่ต้องใช้ในการซื้อขาย” เป็น

$$\text{Days}_i = \frac{V|\Delta w_i|}{\kappa_i ADV_i}$$

ภายใต้สมมติฐานว่า ADV_i และ participation rate คงที่ระหว่างช่วงดำเนินการ

ประเด็นที่ต้องระวัง

Liquidity constraint ควรใช้กับ trade size ไม่ใช่เพียง target weight เพราะต้นทุนและความสามารถในการดำเนินการขึ้นกับขนาดการเปลี่ยน position จาก holdings เดิม

9.12 Covariance Estimation, Shrinkage และ Conditioning

sample covariance matrix คือ

$$S = \frac{1}{T-1} \sum_{t=1}^T (R_t - \bar{R})(R_t - \bar{R})^T$$

เมื่อ n มีขนาดใหญ่เทียบกับ T เมทริกซ์อาจประมาณได้ไม่เสถียร และถ้า $n \geq T$ sample covariance จะมี rank ไม่เกิน $T-1$ จึง singular

แนวทางหนึ่งคือ covariance shrinkage

$$\hat{\Sigma}_\delta = (1-\delta)S + \delta F$$

เมื่อ

$$0 \leq \delta \leq 1$$

และ F เป็น structured target

ทฤษฎีบท 9.3

ถ้า $S \succeq 0$, $F \succeq 0$ และ $0 \leq \delta \leq 1$ แล้ว

$$\hat{\Sigma}_\delta \succeq 0$$

พิสูจน์. สำหรับ $\mathbf{x}$ ใด ๆ

$$\mathbf{x}^T \hat{\Sigma}_\delta \mathbf{x} = (1-\delta)\mathbf{x}^T S \mathbf{x} + \delta \mathbf{x}^T F \mathbf{x} \geq 0$$

□

แนวคิด shrinkage ใช้เพื่อลด estimation variance โดยยอมรับ bias บางส่วน และมีบทบาทสำคัญใน high-dimensional portfolio optimization [30] ถ้า $F \succ 0$ และ $\delta > 0$ จะได้ $\hat{\Sigma}_\delta \succ 0$ แม้ S จะ singular ซึ่งช่วยให้การแก้ระบบเชิงเส้นและ inverse-based calculations มีเสถียรภาพมากขึ้น อย่างไรก็ตาม shrinkage intensity ที่มากเกินไปอาจลบโครงสร้าง covariance ที่มีสาระ จึงควรเลือกด้วยหลักสถิติหรือ validation ไม่ใช่เพื่อทำให้เมทริกซ์กลับด้านได้เพียงอย่างเดียว

9.12.1 Condition Number

ถ้า

$$0 < \lambda_{\min}(\Sigma) \leq \lambda_{\max}(\Sigma)$$

condition number แบบ spectral คือ

$$\kappa(\Sigma) = \frac{\lambda_{\max}(\Sigma)}{\lambda_{\min}(\Sigma)}$$

ค่า κ สูงบ่งชี้ว่า inverse-based optimizer อาจไวต่อ perturbations ถ้า Σ singular จะมี $\lambda_{\min} = 0$ และ condition number ในความหมายนี้เป็นอนันต์ จึงไม่ควรตีความ inverse โดยตรงก่อน regularization หรือการลดมิติ

9.13 Estimation Risk และ Error Maximization

mean-variance optimizer มีแนวโน้มให้น้ำหนักสูงกับสินทรัพย์ที่ estimated return สูง estimated risk ต่ำ หรือ estimated diversification benefit สูง

ถ้าค่าประมาณดังกล่าวเกิดจาก noise optimizer อาจขยาย estimation errors ไปเป็น extreme weights

$$\hat{\mu} = \mu + \varepsilon_{\mu}$$

และ

$$\hat{\Sigma} = \Sigma + \mathcal{E}_{\Sigma}$$

จึงทำให้

$$\hat{w}^* = g(\hat{\mu}, \hat{\Sigma})$$

แตกต่างจาก

$$\mathbf{w}^* = g(\mu, \Sigma)$$

อย่างมากได้

Sensitivity Analysis

สำหรับ parameter θ สามารถตรวจ finite-difference sensitivity

$$\frac{\mathbf{w}^*(\theta + h) - \mathbf{w}^*(\theta - h)}{2h}$$

หรือเปรียบเทียบ

$$\|\mathbf{w}^{(1)} - \mathbf{w}^{(0)}\|_1$$

$$\|\mathbf{w}^{(1)} - \mathbf{w}^{(0)}\|_2$$

และ

$$\max_i |w_i^{(1)} - w_i^{(0)}|$$

ภายใต้ mean, volatility, correlation และ estimation-window perturbations

สำหรับ constrained optimizer ต้องระวังว่า $\mathbf{w}^*(\theta)$ อาจไม่ differentiable ณ จุดที่ active set เปลี่ยน ดังนั้น finite-difference sensitivity ควรถูกตีความเป็น local diagnostic ไม่ใช่อนุพันธ์ที่มีอยู่ทุกจุด การรายงานทั้ง weight change, turnover และ objective deterioration มักให้ภาพความแปรปรวนที่ครบกว่าดู gradient เพียงค่าเดียว

9.14 Black-Litterman: จาก Equilibrium สู่ Views

ปัญหาหนึ่งของ mean-variance optimization คือ expected returns ประมาณได้ยาก Black-Litterman framework เริ่มจาก equilibrium-implied returns แล้วผสมกับ views ที่มี uncertainty

ให้ market weights เป็น $\mathbf{w}_m$ และ risk-aversion parameter เป็น δ ภายใต้ reverse-optimization convention ของ Black-Litterman equilibrium-implied excess returns เขียนได้เป็น

$$\Pi = \delta \Sigma w_m$$

ให้ views อยู่ในรูป

$$P\mu = q + \varepsilon$$

โดยในส่วนนี้ μ และ q ต้องใช้ convention เดียวกับ Π กล่าวคือถ้า Π เป็น excess returns views ก็ควรถูกเขียนบน excess-return scale เช่นกัน

โดย

$$\varepsilon \sim (0, \Omega)$$

ภายใต้ Gaussian formulation ที่ใช้กันทั่วไป posterior mean สามารถเขียนเป็น

Black-Litterman Posterior Mean

$$\mu_{BL} = [(\tau \Sigma)^{-1} + P^T \Omega^{-1} P]^{-1} [(\tau \Sigma)^{-1} \Pi + P^T \Omega^{-1} q]$$

เมื่อ $\tau > 0$ ควบคุม uncertainty ของ prior mean และ Ω เป็น covariance ของ view errors สูตรนี้สมมติ Gaussian prior/view structure และเมทริกซ์ที่เกี่ยวข้องสามารถแก้ระบบเชิงเส้นได้ ในการคำนวณจริงควรใช้ linear solver หรือ factorization ที่เสถียรกว่าการสร้าง matrix inverse แบบตรง ๆ

μ_{BL} เป็น posterior mean ไม่ใช่ realized return ที่รับประกัน และ ตำรา หรือ ซอฟต์แวร์ บางชุดใช้ convention ต่างกันในการนำ posterior uncertainty ไปปรับ covariance ที่ใช้ใน optimization จึงต้องบันทึกให้ชัดว่า optimizer ใช้ Σ เดิมหรือ posterior-adjusted covariance

ประเด็นที่ต้องระวัง

Black-Litterman ไม่ได้กำจัด estimation risk แต่ทำให้ assumptions เกี่ยวกับ prior, views และ confidence ถูกเขียนอย่างชัดเจนขึ้น ผลลัพธ์ยังขึ้นกับ δ, τ, P, q และ Ω

9.15 Robust Optimization

แทนที่จะถือว่า μ และ Σ ทราบแน่นอน robust optimization กำหนด uncertainty set

$$\mu \in \mathcal{U}_\mu$$

และอาจมี

$$\Sigma \in \mathcal{U}_\Sigma$$

ตัวอย่าง robust mean–variance problem คือ

$$\max_{w \in \mathcal{F}} \min_{\mu \in \mathcal{U}_\mu} \left\{ \mu^T w - \frac{\lambda}{2} w^T \Sigma w \right\}$$

แนวคิดคือเลือกพอร์ตที่ยังยอมรับได้เมื่อ inputs เคลื่อนภายใน uncertainty set แทนการ optimize กับ point estimates เพียงชุดเดียว ตัวอย่างที่เห็นกลไกได้ชัดคือ ellipsoidal mean-uncertainty set

$$\mathcal{U}_\mu = \{ \hat{\mu} + u : u^T Q^{-1} u \leq \rho^2 \}, \quad Q \succ 0$$

ซึ่งให้ worst-case expected return

$$\min_{\mu \in \mathcal{U}_\mu} \mu^T w = \hat{\mu}^T w - \rho \sqrt{w^T Q w}$$

จึงเห็นว่า robustness ทำหน้าที่คล้าย penalty ต่อ exposure ที่ไวต่อความไม่แน่นอนของ expected-return estimates โดยตรง

ข้อสังเกต

Robustness ไม่ได้หมายถึงการทำให้พอร์ต conservative มากที่สุด แต่หมายถึงการกำหนด uncertainty อย่างสมเหตุผลแล้วหาคำตอบที่ไม่พึ่งค่าประมาณเฉพาะจุดมากเกินไป

9.16 Solver Validation และ Numerical Diagnostics

การได้เวกเตอร์น้ำหนักจาก software ไม่เพียงพอ ต้องตรวจคำตอบอย่างน้อยดังนี้

ตารางที่ 9.1: รายการตรวจสอบคำตอบของ Portfolio Optimizer

รายการ	สิ่งที่ตรวจ	การตีความ/เกณฑ์ตรวจสอบ
Solver status	status และ termination reason	ยอมรับเฉพาะสถานะที่เหมาะสมกับ solver; ไม่ใช่ weights โดยอัตโนมัติ
Primal feasibility	budget, target, bounds, group, leverage	residual/violation ต้องอยู่ภายใน tolerance ของปัญหา
Covariance	symmetry, PSD, condition number	negative eigenvalue ที่น้อยหรือ near-singularity ต้องได้รับการวิเคราะห์/แก้ไข
Objective recompute	return, variance, Sharpe, TE, costs	คำนวณซ้ำจาก weights เพื่อจับ unit และ coding errors
KKT/dual	stationarity, dual feasibility, complementarity	ใช้ diagnostic ที่ตรงกับ convex/non-smooth formulation
Units/scaling	μ, Σ, R_f , costs	horizon, annualization และหน่วยต้องสอดคล้องกัน
Concentration	max weight, HHI, group/factor exposure	ต้องสอดคล้อง mandate และเหตุผลเชิงเศรษฐศาสตร์
Turnover/liquidity	$\ w - w^{old}\ _1$, trade size, participation	ต้องอยู่ใน limits และดำเนินการได้จริง
Sensitivity	weights, active set, objective เมื่อ input เปลี่ยน	ระบุ unexplained jumps หรือความแปรปรวนของคำตอบ

9.16.1 Numerical Tolerance

คอมพิวเตอร์อาจคืนค่าเช่น

$$w_i = -10^{-10}$$

ในปัญหา long-only ซึ่งอาจเป็น numerical tolerance ไม่ใช่ economically meaningful short position

การตัดสินใจว่าจะปิดค่าเป็นศูนย์ต้องอาศัย solver tolerance และ verification ไม่ใช่กฎตายตัว และหลังการปิดค่าต้องตรวจ budget, bounds และ objective อีกครั้ง เพราะการแก้ post-processing อาจทำให้คำตอบเดิมไม่ feasible หรือไม่ optimal

9.17 ตัวอย่างเชิงคำนวณแบบครบวงจร

พิจารณาสินทรัพย์ 4 ชนิด

$$\mu = \begin{bmatrix} 0.07 \\ 0.10 \\ 0.13 \\ 0.09 \end{bmatrix}$$

volatility vector

$$\sigma = \begin{bmatrix} 0.08 \\ 0.14 \\ 0.20 \\ 0.11 \end{bmatrix}$$

และ correlation matrix

$$P = \begin{bmatrix} 1 & 0.25 & 0.10 & 0.20 \\ 0.25 & 1 & 0.45 & 0.30 \\ 0.10 & 0.45 & 1 & 0.35 \\ 0.20 & 0.30 & 0.35 & 1 \end{bmatrix}$$

สร้าง covariance matrix ด้วย

$$\Sigma = D_{\sigma} P D_{\sigma}, \quad D_{\sigma} = \text{diag}(\sigma)$$

จึงได้

$$\Sigma = \begin{bmatrix} 0.00640 & 0.00280 & 0.00160 & 0.00176 \\ 0.00280 & 0.01960 & 0.01260 & 0.00462 \\ 0.00160 & 0.01260 & 0.04000 & 0.00770 \\ 0.00176 & 0.00462 & 0.00770 & 0.01210 \end{bmatrix}$$

เมทริกซ์นี้เป็น positive definite สำหรับค่าที่กำหนด จึงทำให้ minimum-variance objective เป็น strictly convex

สำหรับ long-only minimum variance ผลเชิงตัวเลขจะให้น้ำหนักประมาณ

$$\mathbf{w}_{MV} \approx \begin{bmatrix} 0.6394 \\ 0.0800 \\ 0.0190 \\ 0.2616 \end{bmatrix}$$

และมี

$$\mathbf{1}^T \mathbf{w}_{MV} = 1, \quad \mu_p \approx 7.88\%, \quad \sigma_p \approx 6.93\%$$

น้ำหนักทุกตัวเป็นบวก จึงหมายความว่า long-only constraints ยังไม่ active ในคำตอบนี้ และ constrained solution ตรงกับ unconstrained GMV solution สำหรับข้อมูลชุดนี้

ถ้ากำหนด target return

$$\mu_0 = 10\%$$

จะได้พอร์ต long-only target-return โดยประมาณ

$$\mathbf{w}_{10\%} \approx \begin{bmatrix} 0.1739 \\ 0.2156 \\ 0.2831 \\ 0.3274 \end{bmatrix}$$

พร้อม

$$\mathbf{1}^T \mathbf{w}_{10\%} = 1, \quad \mu^T \mathbf{w}_{10\%} \approx 10.00\%, \quad \sigma_p \approx 9.90\%$$

เมื่อใช้ค่าที่ไม่พิเศษจะได้ $\sigma_p \approx 9.8955\%$ และน้ำหนักทั้งสี่ยังคงเป็นบวก ดังนั้น target-return constraint เป็นตัวกำหนดตำแหน่งบน frontier แต่ long-only constraints ยังไม่ bind ตัวอย่างนี้จึงเหมาะสำหรับฝึกตรวจ active constraints, budget identity และการคำนวณซ้ำทั้งใน spreadsheet และ numerical solver ไม่ใช่เพียงคัดลอก weights

9.18 กรณีศึกษางานวิจัยและการเชื่อมสู่การใช้งาน

งานเชิงประจักษ์เกี่ยวกับ quantitative portfolio construction แสดงการเปรียบเทียบ risk models และ diversification strategies เช่น minimum variance, maximum diversification และ risk parity พร้อม covariance shrinkage และ position upper bounds ในการทดสอบนอกตัวอย่าง

จากงานวิจัยสู่การประยุกต์: สิ่งที่เราควรเรียนรู้จากงานเชิงประจักษ์

สาระสำคัญสำหรับตำราไม่ใช่นำผลตอบแทนย้อนหลังของกลยุทธ์หนึ่งมาเป็นคำรับรองผลในอนาคต แต่คือการแยก

risk model $\rightarrow$ optimization rule $\rightarrow$ constraints

$\rightarrow$ out-of-sample test $\rightarrow$ robustness

ออกจากกันอย่างชัดเจน

เมื่อเปลี่ยน covariance estimator แม้ใช้ optimization objective เดิม น้ำหนักและ realized risk สามารถเปลี่ยนได้มาก

ข้อสรุปเชิงวิธีวิจัยคือการเปรียบเทียบกลยุทธ์ต้องเปลี่ยนองค์ประกอบที่บางส่วนให้ตรวจสอบได้ เช่น ตรึง objective แล้วเปลี่ยน covariance estimator หรือตรึง risk model แล้วเปลี่ยน constraints การเปลี่ยนหลายองค์ประกอบพร้อมกันทำให้ไม่สามารถระบุได้ว่าผลลัพธ์ที่ดีขึ้นหรือแย่ลงเกิดจากส่วนใดของกระบวนการ

9.19 Out-of-Sample Portfolio Optimization

กระบวนการที่เหมาะสมต้องรักษาลำดับเวลา

$$\mathcal{D}_t \rightarrow (\hat{\mu}_t, \hat{\Sigma}_t) \rightarrow \mathbf{w}_t^* \rightarrow R_{p,t+1}$$

โดย $\mathcal{D}_t$ มีเฉพาะข้อมูลที่ทราบ ณ วันที่ตัดสินใจ

ถ้า rebalance ทุก h งวด

$$\mathbf{w}_t^* = g(\mathcal{D}_t)$$

และ realized gross portfolio return คือ

$$R_{p,t+1:t+h}^{\text{gross}}$$

จากข้อมูลหลังวันตัดสินใจ หากต้นทุนที่เกิด ณ เวลา rebalance ถูกแปลงเป็นสัดส่วนของมูลค่าพอร์ต สามารถรายงานผลตอบแทนสุทธิโดยประมาณเป็น

$$R_{p,t+1:t+h}^{\text{net}} = R_{p,t+1:t+h}^{\text{gross}} - C_t$$

โดยต้องใช้ cost convention เดียวกัน ทุกกลยุทธ์ และ ถ้าใช้ holding periods ที่ทับซ้อนกัน ต้องระวัง dependence ของ observations เมื่อนำผลไปทดสอบเชิงสถิติ

Benchmark Ladder

ก่อนสรุปว่า optimizer เพิ่มคุณค่า ควรเปรียบเทียบอย่างน้อยกับ

1. equal weight
2. current portfolio หรือ policy portfolio
3. long-only minimum variance
4. simple inverse-volatility portfolio
5. strategy ที่ซับซ้อนขึ้นตามวัตถุประสงค์

ทุกกลยุทธ์ ต้องใช้ ข้อมูล ณ เวลาเดียวกัน rebalance schedule เดียวกัน และ transaction-cost assumptions เดียวกัน

9.20 Investment Lab: จาก Spreadsheet สู่ Reproducible Optimizer

Investment Lab: Excel Solver: Constrained Portfolio Optimization

สร้าง workbook อย่างน้อย 7 ซีต

1. **Prices** สำหรับ adjusted prices
2. **Returns** สำหรับ aligned returns
3. **Inputs** สำหรับ $\hat{\mu}$, $\hat{\Sigma}$, R_f และ constraints
4. **Optimization** สำหรับ weights, return, variance, volatility และ Sharpe ratio
5. **Frontier** สำหรับ target-return grid
6. **Trading** สำหรับ current weights, trades, turnover และ costs

7. **Checks** สำหรับ PSD, budget, bounds, target return และ sensitivity
เปรียบเทียบ minimum variance, target return และ maximum Sharpe ภายใต้ long-only และ position caps จากนั้นเปลี่ยน estimation window แล้ววัด weight instability

Investment Lab: Python: Portfolio Optimization Validation

สร้างหรือปรับโมเดลให้มีฟังก์ชันอย่างน้อย

- `check_covariance(cov)`
- `minimum_variance(cov, constraints)`
- `target_return(mu, cov, target, constraints)`
- `maximum_sharpe(mu, cov, rf, constraints)`
- `risk_contributions(weights, cov)`
- `risk_parity(cov, budgets)`
- `transaction_cost(trade, parameters)`
- `solver_diagnostics(result)`
- `rolling_optimization(...)`

unit tests ควรตรวจ budget identity, PSD, target-return feasibility, KKT residuals, Euler risk decomposition, transaction-cost sign และ absence of look-ahead

9.21 ข้อจำกัดและหลักการตีความอย่างรับผิดชอบ

ตารางที่ 9.2: ความเสี่ยงของ Portfolio Optimization และแนวทางควบคุม

ความเสี่ยง	ผลที่อาจเกิด	แนวทางควบคุม
Mean estimation error	extreme weights, error maximization	shrinkage, views, bounds, robust sets
Covariance error	diversification/risk contribution ไม่เสถียร	shrinkage, factor risk model, stress tests
Ill-conditioning	perturbation เล็กถูกขยายผ่าน inverse	eigenvalue/condition diagnostics, regularization
Objective mismatch	optimum ทางคณิตศาสตร์ไม่ตรง mandate	ระบุ investment objective และ loss function ให้ชัด
Infeasible constraints	ไม่มีพอร์ตที่ทำตามเงื่อนไขพร้อมกัน	feasible-range check, constraint audit
Over-concentration	model risk กลายเป็น position risk	position/group/factor caps, concentration metrics
Turnover/cost error	gross alpha ถูกต้นทุนกิน	calibrated costs, turnover control, net test
Liquidity	target trade ดำเนินการไม่ได้ตามเวลา	trade-size, participation, days-to-liquidate limits
Nonstationarity	historical optimum เสื่อมเมื่อ regime เปลี่ยน	rolling estimation, stress/regime diagnostics
Look-ahead bias	backtest สูงเกินจริง	point-in-time data, decision clock, audit trail
Solver/formulation misuse	คำตอบผิดชนิดหรือ residual สูง	ตรวจ convexity, scaling, status, KKT/dual diagnostics

บทสรุปท้ายบท

Portfolio optimization เป็นขั้นตอนที่แปลงแบบจำลองผลตอบแทนและความเสี่ยงให้เป็นการตัดสินใจเชิงน้ำหนักร

minimum-variance problem พื้นฐานคือ

$$\min_w \frac{1}{2} \mathbf{w}^T \Sigma \mathbf{w}$$

ภายใต้ constraints ที่กำหนด

เมื่อ $\Sigma \succeq 0$ และ constraints เป็น convex ปัญหา minimum variance เป็น convex optimization problem และ KKT conditions มีบทบาทสำคัญในการตรวจสอบ optimality

target-return problem เพิ่ม $\mu^T \mathbf{w} \geq \mu_0$ และต้องตรวจสอบว่า target อยู่ใน feasible return interval ก่อนแก้ optimization

ขณะที่ maximum-Sharpe problem ใช้

$$\frac{\mu^T \mathbf{w} - R_f}{\sqrt{\mathbf{w}^T \Sigma \mathbf{w}}}$$

และต้องเลือก formulation ให้เหมาะกับ constraints

การกระจายความเสี่ยงสามารถมองผ่าน Diversification Ratio และ risk contributions

$$RC_i = \frac{w_i (\Sigma \mathbf{w})_i}{\sigma_p}$$

โดย

$$\sum_i RC_i = \sigma_p$$

พอร์ตที่ใช้จริงต้องเพิ่ม position limits, group limits, leverage, tracking error, turnover, transaction costs และ liquidity constraints

ความท้าทายสำคัญไม่ได้อยู่ที่ solver เพียงอย่างเดียว แต่อยู่ที่ estimation risk โดยเฉพาะ $\hat{\mu}$ และ $\hat{\Sigma}$ การใช้ shrinkage, Black-Litterman, robust optimization, constraints และ sensitivity analysis ช่วยลดความแปรปรวนของคำตอบได้ แต่ไม่สามารถกำจัด model risk ทั้งหมด

การประเมิน optimizer จึงต้องใช้ rolling out-of-sample design และ benchmark ที่เหมาะสม โดยรักษา point-in-time information และหักต้นทุนการซื้อขายอย่างสอดคล้อง

บทถัดไปจะนำพอร์ตที่ได้จาก optimization ไปสู่ปัญหาเชิงนโยบายการลงทุน ได้แก่ strategic/tactical asset allocation และ rebalancing

คำศัพท์สำคัญ

- Portfolio optimization
- Decision variable
- Objective function
- Feasible set
- Convex set
- Convex function
- Quadratic programming
- KKT conditions
- Active constraint
- Shadow price
- Minimum variance
- Target return
- Maximum Sharpe
- Diversification ratio
- Maximum diversification
- Marginal risk contribution
- Component risk contribution
- Risk parity
- Risk budget
- Position limit
- Gross exposure
- Tracking error
- Turnover
- Transaction cost
- Liquidity constraint
- Covariance shrinkage
- Condition number
- Estimation risk
- Black-Litterman
- Robust optimization
- Solver tolerance
- Out-of-sample validation

แบบฝึกหัดท้ายบท

1. แยก decision variables, objective, constraints และ inputs ของ long-only minimum-variance problem
2. พิสูจน์ว่า $\mathbf{w}^T \Sigma \mathbf{w}$ เป็น convex เมื่อ $\Sigma \succeq 0$
3. อธิบายความแตกต่างระหว่าง positive semidefinite กับ positive definite ในบริบทของ portfolio optimization
4. เขียน KKT conditions ของ long-only minimum-variance problem
5. อธิบาย complementary slackness สำหรับสินทรัพย์ที่มี $w_i = 0$
6. เขียน target-return minimum-variance problem ที่มี $0 \leq w_i \leq 0.30$
7. พิสูจน์ว่า long-only fully invested expected return อยู่ระหว่าง minimum และ maximum ของ asset expected returns
8. สร้างตัวอย่าง target return ที่ infeasible
9. อธิบายเหตุใด constrained frontier ไม่สามารถอยู่เหนือ unconstrained opportunity set ในความ

หมายของ minimum variance ที่ target เดียวกัน

10. อธิบายว่าเหตุใด maximum-Sharpe problem ไม่ใช่ quadratic program โดยตรง
11. เขียน Diversification Ratio และอธิบาย numerator กับ denominator
12. หา MRC_i , RC_i และ PRC_i ของพอร์ตที่กำหนด
13. พิสูจน์ Euler risk decomposition
14. อธิบายความแตกต่างระหว่าง equal weight, inverse volatility และ risk parity
15. เขียน portfolio problem ที่มี sector cap 25%
16. อธิบายความแตกต่างระหว่าง net exposure กับ gross exposure
17. เขียน tracking-error constraint เทียบ benchmark
18. เปรียบเทียบ two-way turnover กับ one-way turnover conventions
19. สร้าง objective ที่รวม expected return, variance และ linear transaction cost
20. อธิบายเหตุผลที่ liquidity constraint ควรผูกกับ trade vector
21. พิสูจน์ว่า shrinkage covariance ยังคง PSD เมื่อ matrices ต้นทางเป็น PSD
22. อธิบายเหตุผลที่ sample covariance singular เมื่อ $n \geq T$ ภายใต้ข้อมูลที่ demean แล้ว
23. อธิบายผลของ condition number สูงต่อ inverse-based portfolio weights
24. ออกแบบ sensitivity test ต่อ expected returns ± 50 basis points
25. อธิบายแนวคิด error maximization ของ mean-variance optimizer
26. คำนวณ equilibrium-implied returns $\Pi = \delta \Sigma \mathbf{w}_m$ จากข้อมูลที่กำหนด
27. อธิบายบทบาทของ P , q และ Ω ใน Black-Litterman
28. เขียน robust optimization problem เมื่อ expected returns อยู่ใน uncertainty set
29. ออกแบบ solver-validation checklist สำหรับ constrained portfolio
30. ออกแบบ rolling out-of-sample experiment เปรียบเทียบ equal weight, minimum variance และ maximum Sharpe หลัง transaction costs
31. อธิบายว่าเหตุใดข้อจำกัด $\mu^T \mathbf{w} \geq \mu_0$ อาจไม่ bind เมื่อ μ_0 ต่ำกว่าผลตอบแทนของ constrained GMV และเปรียบเทียบกับการใช้ equality
32. ยกตัวอย่างสถานการณ์ที่ $RC_i < 0$ แล้วอธิบายความหมายเชิง hedge และผลต่อการตีความ risk budget

คำถามชวนคิด

1. ถ้า optimizer ให้คำตอบที่มี Sharpe ratio สูงมากแต่เปลี่ยนน้ำหนักเกือบทั้งหมดเมื่อ expected returns เปลี่ยนเพียงเล็กน้อย ควรเชื่อคำตอบนั้นเพียงใด
2. การเพิ่ม constraints เป็นการทำลาย optimality หรือเป็นการนิยาม optimality ให้สอดคล้องกับโลกจริงมากขึ้น
3. หาก expected returns พยากรณ์ได้ไม่ดี แต่ covariance matrix ค่อนข้างเสถียร กลยุทธ์ใดควรใช้เป็น benchmark
4. Black-Litterman แก้ปัญหา estimation risk หรือเพียงจัดโครงสร้าง assumptions ให้ดีขึ้น
5. เหตุใดพอร์ตที่ชนะใน sample จึงอาจแพ้ equal-weight portfolio นอก sample
6. ในยุคที่ optimization software ใช้งานง่าย ความรู้เรื่อง convexity, KKT และ numerical diagnostics ยังจำเป็นหรือไม่

เอกสารอ้างอิงท้ายบทที่ 9

- [7] Fischer Black and Robert Litterman. “Global Portfolio Optimization”. In: *Financial Analysts Journal* 48.5 (1992), pp. 28–43.
- [30] Olivier Ledoit and Michael Wolf. “A Well-Conditioned Estimator for Large-Dimensional Covariance Matrices”. In: *Journal of Multivariate Analysis* 88.2 (2004), pp. 365–411.
- [33] Harry Markowitz. “Portfolio Selection”. In: *The Journal of Finance* 7.1 (1952), pp. 77–91. DOI: [10.2307/2975974](https://doi.org/10.2307/2975974).

