Chapter. 05

리스크 관리

현대 포트폴리오 이론 (MPT) III

FASTCAMPUS
ONLINE
금융공학/퀀트

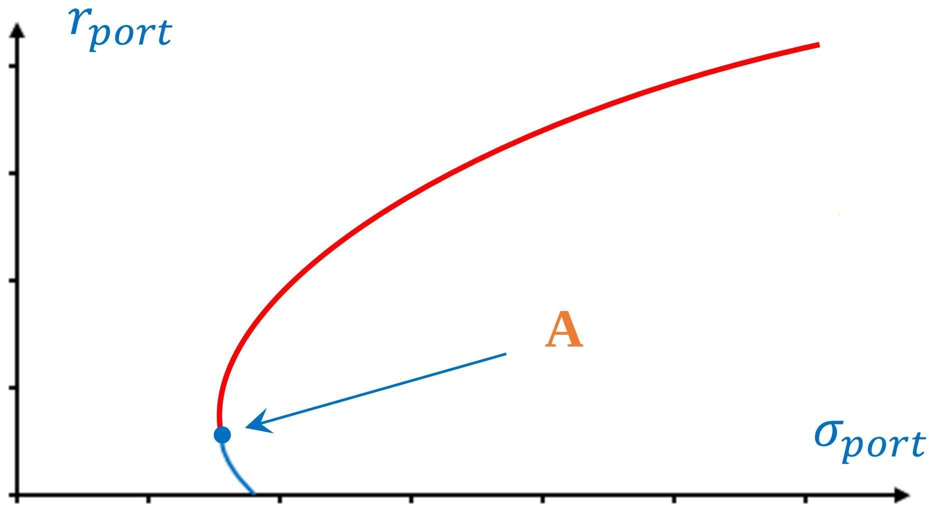
강사. 장순용

l 키포인트

- 효율적 투자선 (Efficient frontier).
- 두 개의 펀드 이론 (Two fund theorem).
- Long 포지션에 국한된 포트폴리오 (난이도 上, option).
- Quadratic programming (난이도 上, option).

I 효율적 투자선 (Efficient frontier)

- r_{port}를 정해놓고 σ_{port}를 최적화하여 다음 그래프를 그릴 수 있다.

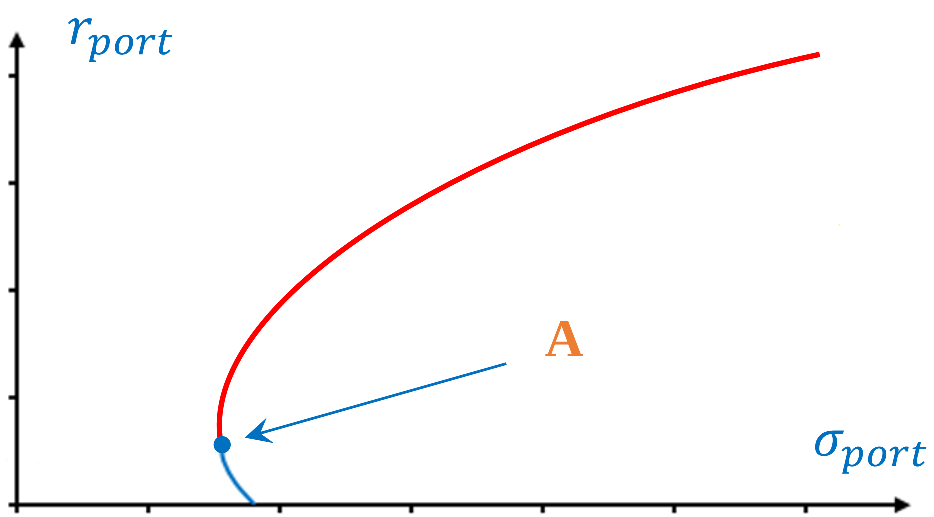


$\Rightarrow$ 곡선의 상위 부분 (적색) 을 "효율적 투자선"이라 부른다.

$\Rightarrow$ 목표하는 수익률 대비 최소의 리스크 지점이다.

I 효율적 투자선 (Efficient frontier)

- r_{port}를 정해놓고 σ_{port}를 최적화하여 다음 그래프를 그릴 수 있다.



$\Rightarrow$ 매번 r_{port}를 조금씩 변경하고 최적화를 실시해야 하는 번거로움이 따른다.

I 두 개의 펀드 이론 (Two fund theorem)

- 먼저 효율적 투자선上의 두 개의 좌표 (포트폴리오)를 전제 한다.

예). 두 개의 좌표는 A 포인트와 T 포인트가 될 수 있다.

$\Rightarrow$ (σ_1, r_1)과 (σ_2, r_2)로 표기하기로 한다.

$\Rightarrow$ 각각의 가중치 벡터는 $\overrightarrow{w_1}$과 $\overrightarrow{w_2}$로 표기한다.

$\Rightarrow$ 효율적 투자선의 다른 좌표는 $\overrightarrow{w_1}$과 $\overrightarrow{w_2}$ 의 선형 조합으로 얻을 수 있다.

$$\overrightarrow{w} = (1 - \alpha)\,\overrightarrow{w_1} + \alpha\,\overrightarrow{w_2}$$

$\Rightarrow$ α 를 변경시키며 σ와 r을 계산해서 효율적 투자선을 그릴 수 있다. (최적화 불필요!)

I Long 포지션에 국한된 포트폴리오 (난이도 上)

- 지금까지는 가중치 w_i가 양수 (+) 또는 음수 (-)가 될 수 있었다.

$\Rightarrow$ 가중치 w_i가 양수이면 Long (매수) 포지션을 의미한다.

$\Rightarrow$ 가중치 w_i가 음수이면 Short (매도) 포지션을 의미한다.

- 그런데 주식과 같은 자산을 Short (공매도) 포지션을 두기에는 현실적으로 어려울 수도 있다.

$\Rightarrow$ 모든 가중치 w_i가 양수라는 부등식 제약을 두기로 한다: $\vec{w} \geq \vec{0}$

$\Rightarrow$ Long 포지션에 국한된 포트폴리오를 만들고자 한다.

I Quadratic Programming (난이도 上)

- 다음과 같이 두 개의 등식으로 성립되는 제한조건과 한 개의 부등식으로 성립되는 제한조건을 전제한다.

$\Rightarrow$ 수익률 고정: $r_{port} = \sum_{i=1}^{P} w_i r_i$

$\Rightarrow$ 정규화: $\sum_{i=1}^{P} w_i = 1$

$\Rightarrow$ Long 포지션에 국한: $w_i \geq 0$

I Quadratic Programming (난이도 上)

- 라그랑주 함수는 다음과 같다.

$$\mathcal{L} = \frac{1}{2}\vec{w}^t \tilde{\Sigma}\vec{w} - \frac{1}{c}\sum_{i=1}^{N} Log(w_i) - \lambda_1(\vec{w}\cdot\vec{1} - 1) - \lambda_2(\vec{w}\cdot\vec{r} - r_{port})$$

$\Rightarrow$ $-\frac{1}{c}\sum_{i=1}^{N} Log(w_i)$가 바로 부등식에 의한 제약을 나타낸다.

$\Rightarrow$ 파라미터 c가 작을수록 강한 제한조건이 적용되는 것이다.

$\Rightarrow$ 파라미터 c가 클수록 약한 제한조건이 적용되는 것이다.

$\Rightarrow$ 로그함수는 비선형이기 때문에 선형 방정식으로 그 해를 구할 수 없다.

I Quadratic Programming (난이도 上)

- 라그랑주 함수를 다음과 같이 표현할 수 있다.

$$\mathcal{L}(\vec{w},\vec{\lambda}) = f(\vec{w}) - \vec{\lambda}\cdot(\tilde{A}\,\vec{w} - \vec{b})$$

$$f(\vec{w}) = \frac{1}{2}\vec{w}^{t}\tilde{\Sigma}\,\vec{w} - \frac{1}{c}\sum_{i=1}^{N} Log(w_i)$$

$$\tilde{A} = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ r_1 & r_2 & \cdots & r_N \end{pmatrix},\ \vec{b} = \begin{pmatrix} 1 \\ r_{port} \end{pmatrix},\ \vec{\lambda} = \begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix}$$

I Quadratic Programming (난이도 上)

- 비선형 방정식을 단 한번에 풀기는 어렵다.

⇒ 비선형 방정식을 근접하여 선형 방정식化하는 방법을 적용한다.

⇒ 최적화된 해로 조금씩 수렴해 나가는 방식을 적용한다.

⇒ 라그랑주 함수의 최적화 조건은 다음과 같이 함축적으로 표현할 수 있다.

$$\vec{\nabla}_{\vec{w}}\mathcal{L}(\vec{w},\vec{\lambda}) + \tilde{\nabla}^2_{\vec{w}}\mathcal{L}(\vec{w},\vec{\lambda})\overrightarrow{\Delta w} + \tilde{\nabla}^2_{\vec{w},\vec{\lambda}}\mathcal{L}(\vec{w},\vec{\lambda})\overrightarrow{\Delta\lambda} = \vec{0}$$

$$\vec{\nabla}_{\vec{\lambda}}\mathcal{L}(\vec{w},\vec{\lambda}) + \tilde{\nabla}^2_{\vec{w},\vec{\lambda}}\mathcal{L}(\vec{w},\vec{\lambda})\overrightarrow{\Delta w} = \vec{0}$$

⇒ 도출 방법은 부록 참고.

I Quadratic Programming (난이도 上)

- 실제 라그랑주 함수를 대입해서 정리하면 다음과 같은 방정식을 얻을 수 있다. 그리고 $\overrightarrow{\Delta w}$ 와 $\overrightarrow{\Delta \lambda}$가 방정식의 해이다.

$$\begin{pmatrix} \tilde{\Sigma} + \frac{1}{c}\tilde{\Phi} & -\tilde{A}^{t} \\ \tilde{A} & \vec{0} \end{pmatrix} \begin{pmatrix} \overrightarrow{\Delta w} \\ \overrightarrow{\Delta \lambda} \end{pmatrix} = -\begin{pmatrix} \tilde{\Sigma}\,\vec{w} \\ \vec{d}\,(\vec{w}) \end{pmatrix}$$

$$\tilde{\Phi} = \begin{pmatrix} \frac{1}{(w_1)^2} & 0 & \cdots & 0 \\ 0 & \frac{1}{(w_2)^2} & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & \frac{1}{(w_P)^2} \end{pmatrix} \quad , \quad \vec{d}\,(\vec{w}) = \tilde{A}\,\vec{w} - \vec{b}$$

| Quadratic Programming (난이도 上) : 알고리즘

1. 모든 제한조건을 충족하는 $\vec{w}$와 $\vec{\lambda}$의 초기값을 "선택"한다. 또한, 파라미터 c에 임의의 작은 값을 부여한다. 이것은 강한 부등식 제한조건을 적용하는 것을 의미한다.

2. 방정식의 해 $\overrightarrow{\Delta w}$ 와 $\overrightarrow{\Delta \lambda}$를 구한다.

3. $\vec{w} \rightarrow \vec{w} + \overrightarrow{\Delta w}$ 그리고 $\vec{\lambda} \rightarrow \vec{\lambda} + \overrightarrow{\Delta \lambda}$와 같이 최적화 지점으로 한 스텝 수렴한다.

4. 스텝 2에서부터 일정 회수를 반복한다.

I Quadratic Programming (난이도 上) : 알고리즘

5. 수렴된 $\vec{w}$와 $\vec{\lambda}$ 을 초기값으로 하며 파라미터 c의 값을 증가시키며 반복한다.이것은 부등식 제한조건 $\vec{w} \geq \vec{0}$을 조금씩 완화해 가는 것을 의미한다.

$\Rightarrow$ 만약에, 벡터 $\vec{w}$의 성분 중에서 0으로 수렴하는 것이 있다면,

너무 강한 제한조건은 0으로의 수렴을 방해할 수 있기 때문이다.

$\Rightarrow$ 아무리 제한조건이 약해지더라도 $-\frac{1}{c}\sum_{i=1}^{N} Log(w_i)$때문에 절대로

가중치가 0을 지나서 음수 (short)로 넘어가는 상황은 발생하지 않는다.

I 끝.

Life Changing Education

감사합니다.

FAST CAMPUS
ONLINE
장순용 강사.