# 투자론 중간고사 대체과제 (2024년 봄학기)

학번: 202368020
이름: 김재환

**1. 투자전략 (25점)**

최근 경제뉴스를 보면, 주식형 펀드 가운데 인덱스형 펀드의 수익률이 액티브형 펀드의 수익률보다 월등히 좋은 것으로 나타나고 있다.

1) 인덱스형과 액티브형 펀드에 각각 투자하는 행위는 어떤 투자전략으로 볼 수 있는가? (10점)

인덱스형과 액티브형 펀드에 투자하는 것은 두 가지 다른 투자 전략을 나타냅니다.

**인덱스 펀드 투자 (Index Fund Investing):**

투자 전략: 인덱스 펀드는 특정 지수 (예: S&P 500, NASDAQ)를 추적하는 데 초점을 맞춘 투자 전략입니다. 이러한 펀드는 해당 지수에 포함된 주식을 비롯한 자산을 보유하고 해당 지수의 성과를 따라갑니다. 따라서 인덱스 펀드 투자는 시장을 지수 수준에서 따라가는 전략으로 볼 수 있습니다.

**액티브 펀드 투자 (Active Fund Investing):**

투자 전략: 액티브 펀드는 펀드 매니저나 관리팀이 선택한 주식, 채권, 자산 등으로 구성된 포트폴리오를 보유하는 투자 전략입니다. 액티브 펀드 매니저는 시장의 특정 부분에서 상대적인 가치를 찾으려고 노력하고, 주식 선택, 시장 타이밍, 산업 및 기업 분석 등을 통해 초과 수익을 얻으려고 합니다. 따라서 액티브 펀드 투자는 시장을 능동적으로 탐색하여 초과 수익을 추구하는 전략으로 볼 수 있습니다.

2) 인덱스형 펀드의 성과가 우월한 것을 당신은 어떻게 해석하는가? (15점)

인덱스형 펀드의 성과가 우월하다는 것은 해당 펀드가 시장 지수를 따라가는 passively managed fund로서, 해당 지수의 성과를 잘 추적하거나 뛰어넘는 결과를 나타낸다는 것을 의미합니다. 이는 특정 지수의 성과를 달성하는 데 있어서 비용 효율적이고 일관된 결과를 얻을 수 있다는 것을 시사합니다.

인덱스형 펀드는 엑티브형 펀드에 비해 운영수수료가 낮아 저렴한 비용으로 투자를 할 수 있으며, 시장지수를 따라가기 때문에 엑티브형 펀드 보다는 중립적으로 운영된다고 볼 수 있습니다.
또한, 엑티브형 펀드보다 표준편차가 낮아 리스크가 분산되며, 시장 지수를 따르기 때문에 비교적 일관적 성과를 제공합니다.
따라서 장기적인 투자계획을 수립하는데 도움이 될 수 있습니다.

**2. 마코위츠 모형 (25점)**

1) 수업시간에 공부한 마코위츠의 모형에 의하면 KOSPI의 기대수익률은 어떤 방식으로 추정해야 하는가? 2) 이제, 현실에서 당신이 2019년 KOSPI의 기대수익률을 추정해야 한다고 가정하자. 당신은 구체적으로 어떤 방식으로 2019년 KOSPI의 기대수익률을 추정하겠는가?

1) 마코위츠의 포트폴리오 이론에 따르면 KOSPI의 기대수익률은 다음과 같은 방식으로 추정할 수 있습니다:

KOSPI의 기대수익률은 각 주식의 기대수익률과 해당 주식의 시장가치 비중을 곱한 것의 합으로 계산됩니다.
시장가치 비중은 각 주식의 시가총액을 전체 시장의 시가총액으로 나눈 것으로, 주식의 상대적인 중요성을 반영합니다.
마코위츠의 포트폴리오 이론은 최적의 포트폴리오 구성과 관련된 것이고 KOSPI 지수는 시장전체의 지표로 보는 것이 맞으므로 개별주식 혹은 여러 주식을 선택할 때 구성된 포트폴리오의 분산(리스크)과 자산들 간의 상관관계가 적은 것이 포트폴리오 최적화 구성에 적합하다는 이론입니다.
따라서 포트폴리오의 분산 혹은 자산들 간의 상관계수를 직접 계산 시 포트폴리오의 자산 수가 많을 때 계산의 복잡성이 올라가지만, KOSPI지수가 있다면 시장지수와 개별 주식 간의 상관관계만 구하면 되기 때문에 계산의 복잡성을 매우 낮출 수 있습니다.
이론적으로 특정기간의 KOSPI 시장 전체의 개별 주식의 기대수익률을 구하고, 그에 따른 시가총액의 가중치를 반영하고 KOSPI 전체 주식의 포트폴리오를 구성한다면(특정주식을 배제하지 않고 모두 포함), 특정 기간의 KOSPI 지수의 수익률과 동일하게 됩니다.

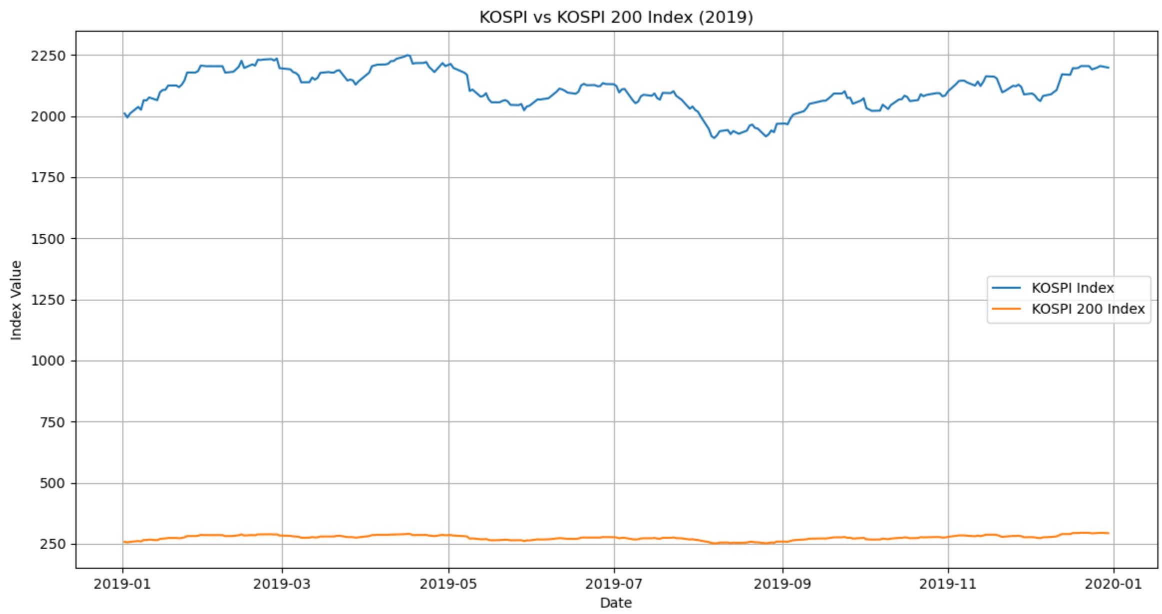
2) 현실에서 2019년 KOSPI의 기대수익률을 구하는 방법은 다음과 같습니다.

각 개별 주식의 2019년 기대수익률을 구하고, KOSPI 시가총액의 자산비중을 반영하여 구할 수 있겠지만, 저는 위에서 언급한 KOSPI 지수의 기대수익률을 구하는 방법을 사용하겠습니다.
KOSPI 지수는 비교시점의 시가총액 합계를 기준시점(1980년 1월 4일)의 시가총액 합계를 나눈값의 100을 곱한 값으로 매일의 주가변동이 지수에 반영되는 특징을 가지고 있습니다.
따라서 매일의 KOSPI 지수의 변동을 측정하고자 하는 기간의 합을 평균을 하면 기대수익률을 얻을 수 있습니다.
파이썬을 사용하여 계산을 하였고 구체적으로 'FinanceDataReader' 라이브러리를 사용하여 계산하였습니다.
구체적인 결과값은 다음과 같습니다. (KOSPI PRICE.py 참고)

2019년 KOSPI 지수의 연간 기대수익률: 9.96%
2019년 KOSPI 200의 연간 기대수익률: 14.17%

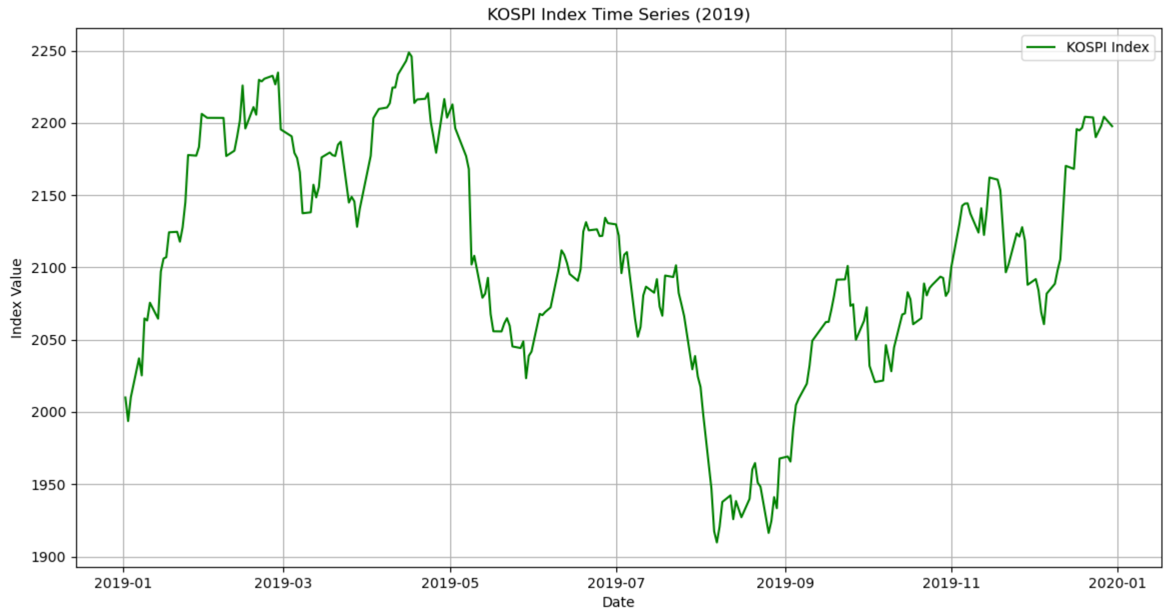
[추가분석]
2019년 초 KOSPI 전체 기업의 수는 901개 였으며, 5분위로 나누었을 때(약 180개 기업) 맨 상위의 그룹의 시가 총액이 89.53%를 차지하고 있습니다. 2019년 말이 되면 KOSPI 전체 기업의 수는 916개로 늘어나고 있으며, 상위 5분위의 시가총액은 90.38%로 늘어나게 됩니다.(현재는 954개 기업이며 상위 5분위는 91.36%로 더 늘어났습니다.) (KOSPIdata.xlsx 참조)
따라서 자연스러운 질문은 KOSPI지수와 KOSPI200지수와의 상관관계 였으며, 두개를 비교하여 파이썬으로 구현하였습니다.

아래는 KOSPI 지수와 KOSPI200지수의 그래프입니다. (Graph of KOSPI Index(FinanceDataReader)_together.py 참고)

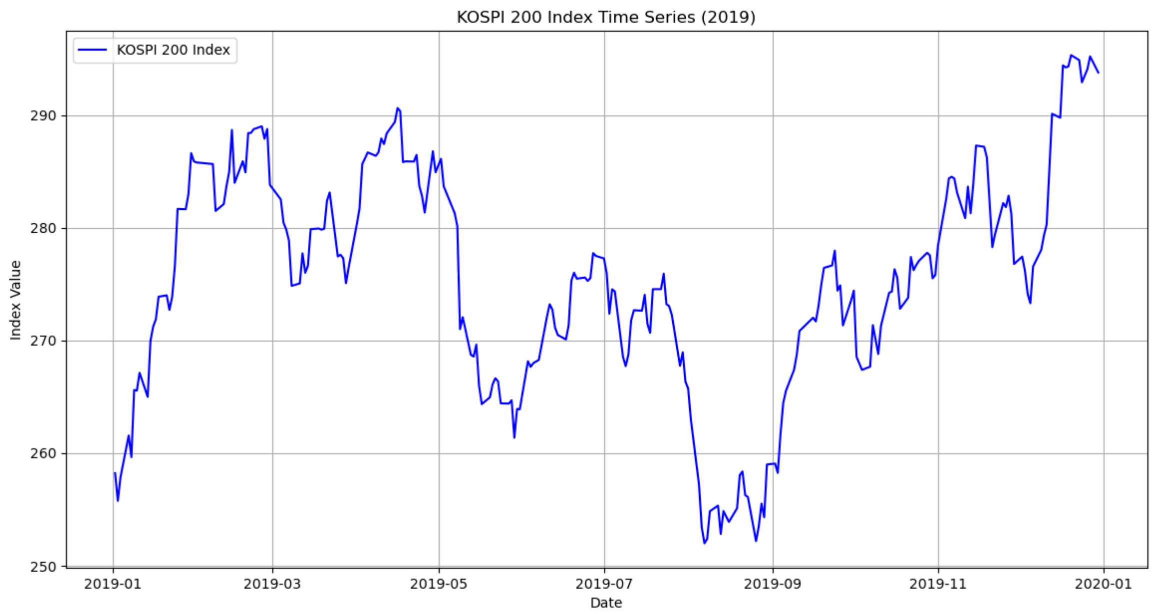



위 시계열 그래프에서는 KOSPI 지수가 KOSPI200 지수보다 변동성이 더 커보이지만 이는 착시적 현상이고 지수의 영역크기가 서로 달라서 발생하는 차이에 불과합니다.
따라서 제대로 된 분석을 위하여 각각의 개별 그래프를 구하였습니다.

아래는 KOSPI 지수의 시계열 그래프입니다.( Graph of KOSPI Index(FinanceDataReader)_alone.py 참고)



다음은 KOSPI200지수의 시계열 그래프입니다. (Graph of KOSPI200 Index(FinanceDataReader.py 참고)




보시다시피, KOSPI 지수와 KOSPI200지수의 그래프 변동이 서로 일치하는 것을 알 수 있습니다. 이는 KOSPI200 지수가 단순히 상위 200개의 기업을 샘플링한 것이 아니라 KOSPI 시장 전체를 반영하도록 종목구성을 해놓은 것으로 추측할 수 있습니다.(실제로 KOSPI200지수 종목의 구성 항목을 보면 상위 뿐만 아니라 중위, 하위 항목들이 포함되어있습니다.)

파이썬을 이용하여 구체적인 수치로 비교하였습니다. 결과는 아래와 같습니다.

(Graph of KOSPI Index(FinanceDataReader)_calculation.py 참고)

KOSPI Index Annual Expected Return: 10.47%
KOSPI Index Annual Variance: 0.015581
KOSPI Index Annual Standard Deviation: 0.124822
KOSPI 200 Index Annual Expected Return: 15.22%
KOSPI 200 Index Annual Variance: 0.018200
KOSPI 200 Index Annual Standard Deviation: 0.134909
Correlation between KOSPI and KOSPI 200: 0.988487
Annual Covariance between KOSPI and KOSPI 200: 0.016646

KOSPI 지수와 KOSPI200지수와의 상관계수는 98.84%이며 리스크 측면에서는 사실상 거의 동일한 항목으로 간주되어집니다. 하지만 KOSPI 200지수의 기대수익률이 4% 이상 높기 때문에 KOSPI 지수와 위험은 동일하지만 기대수익률이 더 높은 종목이라고 할 수 있습니다.

위에서 구한 수익률과 차이가 나는 이유는 두 코드 모두 pct_change() 함수를 사용하여 일간수익률을 구한 후 계산하는 것은 동일하지만, 위에서는 단순 합산한 것의 평균으로 계산하였고, 두번째 방식에서는 같은 방법으로 일간수익률을 구한 다음, 복리로 연간수익률을 계산하였기 때문입니다.

**3. (CAPM, 25점)**
시장포트폴리오의 기대수익률이 연 10%이고, 무위험이자율이 연 4%라고 가정하자. 또한, 주식 A의 시장베타는 1.5라고 가정하자.

1) 주식 A의 시장베타가 1.5인데, 그 의미를 서술하시오. (8점)

주식 A의 시장베타가 1.5 라는 것은 A주식의 수익률이 시장전체(혹은 대표지수종목)의 수익률과 양(+)의 방향으로 시장전체(혹은 대표지수종목) 보다 더 영향을 받는 다는 의미입니다..

즉, 바꿔말하면 해당 주식이 시장의 움직임에 비해 상대적으로 더 큰 변동성을 보인다는 것을 의미합니다. 보통 시장베타는 1로 설정되며, 이보다 높은 시장베타는 해당 주식이 시장의 변동성에 민감하게 반응한다는 것을 나타냅니다.

시장베타가 1.5라는 것은 주식 A가 시장 평균 수익률의 1.5배의 변동성을 보인다는 것을 의미합니다. 따라서 시장이 상승할 때는 주식 A가 상대적으로 더 높은 수익률을 보일 가능성이 있지만, 시장이 하락할 때는 주식 A가 더 큰 손실을 입을 수 있다는 것을 의미합니다.

시장베타는 투자자가 특정 주식의 리스크와 수익률을 평가하는 데 도움을 줍니다. 높은 시장베타를 가진 주식은 시장의 움직임에 민감하게 반응하므로 더 큰 주의가 필요합니다.

2) 이 주식시장에서 시장위험프리미엄은 몇 %인가? (7점)
CAMP의 자산가격결정에서의 수식은 다음과 같습니다.
E(ri) = rf +βi(E(rm) - rf)
위에서 시장프리미엄은 E(rm) – rf 개별주식의 수익률에 무위험이자율을 차감한 것으로 개별주식 매입에 대한 리스크의 보상적 의미를 함축하고 있습니다.
위 식에서의 시장위험프리미엄 10% - 4% = 6% 입니다.

3) CAPM이 성립할 경우, 주식 A의 기대수익률은 얼마인가? (10점)

기대수익률의 공식은 다음과 같습니다.
E(ri) = rf +βi(E(rm) - rf)
이를 대입하면
E(ri) = 4% + 1.5*(10%-4%) = 13%

**4. 최적포트폴리오 선택 (25점)**

1) 경제 내에 두 가지 위험 자산이 주식 A와 주식 B만 존재한다고 하자. 투자자는 1억원을 A주식과 B주식에 나누어 투자하려고 한다. 이 때, 어떤 방식으로 최적 포트폴리오를 선택하게 되는지를 설명하라. 단, 설명과정에서 "투자기회집합"과 "무차별곡선" 이라는 단어를 사용할 것. (13점)

투자자가 주식 A와 주식 B 중에서 어떤 비율로 포트폴리오를 구성할지 결정하기 위해서는 투자 기회 집합과 무차별곡선을 고려해야 합니다.

**투자 기회 집합 (Opportunity Set):** 투자 기회 집합은 투자자가 선택할 수 있는 모든 가능한 포트폴리오의 집합을 나타냅니다. 주식 A와 주식 B의 수익률과 리스크가 주어진다면, 각 주식을 서로 다른 비율로 조합하여 만들 수 있는 다양한 포트폴리오가 투자 기회 집합을 형성합니다.

**무차별곡선 (Indifference Curve):** 무차별곡선은 투자자가 특정 수익률과 리스크 수준에 대해 동일한 만족감을 느끼는 포트폴리오의 집합을 나타냅니다. 즉, 투자자는 무차별곡선 위에 위치한 포트폴리오 중 하나를 선택할 것입니다.

따라서 최적 포트폴리오를 선택하는 과정은 다음과 같습니다:
먼저, 주식 A와 주식 B의 예상 수익률과 리스크를 고려하여 투자 기회 집합을 형성합니다.
다음으로, 투자자의 선호도와 투자 목표를 고려하여 무차별곡선을 결정합니다. 이것은 투자자가 특정 수익률과 리스크 수준에서 동일한 만족감을 느끼는 포트폴리오의 집합입니다.
마지막으로, 무차별곡선과 투자 기회 집합이 만나는 지점이 최적 포트폴리오가 됩니다. 이 지점에서 투자자의 선호도와 투자 기회가 균형을 이룹니다.
결국 최적 포트폴리오를 선택하는 것은 투자자의 리스크 선호도와 수익률 목표에 따라 달라지며, 투자 기회 집합과 무차별곡선을 고려하여 이를 결정합니다.

2) 경제 내에 철수와 영희가 있다고 가정하자. 두 사람은 1)번 문제에서 등장한 주식 A와 주식 B로 구성된 최적의 포트폴리오를 구성하고자 한다. 두 사람이 최종적으로 선택한 최적 포트폴리오는 같은가? 다른가? 그 이유는? (12점)

철수와 영희가 최적의 포트폴리오를 선택하는 과정에서 두 사람이 선택한 포트폴리오가 같을 수도 있고, 다를 수도 있습니다. 이는 두 사람의 투자 선호도, 리스크 태도, 수익률 목표 등에 따라 달라질 수 있습니다.

**1. 같을 수 있는 경우:**
두 사람이 동일한 수익률 목표를 가지고 있고, 동일한 리스크 선호도를 갖고 있다면, 최적의 포트폴리오가 같을 수 있습니다. 이 경우, 두 사람이 무차별곡선과 투자 기회 집합을 고려하여 동일한 포트폴리오를 선택할 수 있습니다.
**2. 다를 수 있는 경우:**
철수와 영희의 수익률 목표나 리스크 선호도가 서로 다르다면, 선택한 포트폴리오가 다를 수 있습니다. 예를 들어, 철수가 더 높은 수익률을 원하고 영희는 더 낮은 리스크를 선호한다면, 두 사람은 서로 다른 포트폴리오를 선택할 것입니다.
또한, 두 사람의 초기 자본금이나 투자 목표 등도 다를 수 있으며, 이러한 요소들이 최종 포트폴리오 선택에 영향을 미칠 수 있습니다.
따라서 철수와 영희가 선택한 최적 포트폴리오가 같을지 다를지는 각자의 투자 선호도와 목표에 따라 달라질 것입니다.

### 3. 유사한 사례

위의 철수와 영희 대신 John(위험 회피가 낮음)과 David(위험 회피가 높음)의 A주식과 B주식의 선호도에 대한 구체적인 사례를 들겠습니다.
먼저, 사례 구현에 앞서 A주식은 기대 수익률은 8%, 표준 편차는 15% 이며, B주식은 기대 수익률은 12%, 표준 편차는 20% 이고, 두 주식 간의 상관관계는 0.5라고 가정합니다.
아래는 파이썬으로 투자기회집합 테이블을 만들고, John과 David의 무차별 곡선을 구현하는 코드입니다.

```
import numpy as np
import matplotlib.pyplot as plt

# 주식 A와 B의 기대 수익률, 표준 편차, 상관계수
mu_A = 0.08
sigma_A = 0.15
mu_B = 0.12
sigma_B = 0.20
correlation = 0.5

# 포트폴리오의 가중치 (0%에서 100% 사이)
weights = np.linspace(0, 1, 100)

# 포트폴리오의 기대 수익률과 표준 편차 계산
portfolio_returns = weights * mu_A + (1 - weights) * mu_B
portfolio_std_devs = np.sqrt((weights * sigma_A)**2 + ((1 - weights) * sigma_B)**2 + 2 * weights * (1 - weights) *
correlation * sigma_A * sigma_B)
portfolio_variances = portfolio_std_devs**2

# 투자기회집합 테이블 생성
for weight, ret, std in zip(weights, portfolio_returns, portfolio_std_devs):
    print(f"Weight A: {weight:.2f}, Weight B: {1-weight:.2f}, Return: {ret:.2%}, Risk: {std:.2%}")

# John과 David의 무차별곡선 설정
def utility_curve(returns, aversion):
    return returns - 0.5 * aversion * (portfolio_std_devs ** 2)

# John (위험 회피가 낮음)
utility_john = utility_curve(portfolio_returns, 2)

# David (위험 회피가 높음)
utility_david = utility_curve(portfolio_returns, 5)

# John과 David의 최적 포트폴리오 찾기
optimal_index_john = np.argmax(utility_john)
optimal_index_david = np.argmax(utility_david)
```

```
# 결과 출력
print()
print(f"John's Optimal Portfolio:")
print(f"   Weight in Stock A: {weights[optimal_index_john]*100:.2f}%")
print(f"   Weight in Stock B: {(1-weights[optimal_index_john])*100:.2f}%")
print(f"   Expected Return: {portfolio_returns[optimal_index_john]*100:.2f}%")
print(f"   Standard Deviation: {portfolio_std_devs[optimal_index_john]:.2f}")
print(f"   Variance: {portfolio_variances[optimal_index_john]:.2f}₩n")

print(f"David's Optimal Portfolio:")
print(f"   Weight in Stock A: {weights[optimal_index_david]*100:.2f}%")
print(f"   Weight in Stock B: {(1-weights[optimal_index_david])*100:.2f}%")
print(f"   Expected Return: {portfolio_returns[optimal_index_david]*100:.2f}%")
print(f"   Standard Deviation: {portfolio_std_devs[optimal_index_david]:.2f}")
print(f"   Variance: {portfolio_variances[optimal_index_david]:.2f}")

# 그래프로 결과 시각화
plt.figure(figsize=(10, 6))
plt.plot(portfolio_std_devs, portfolio_returns, label='Investment Opportunity Set')
plt.plot(portfolio_std_devs, utility_john, '--', label='John₩'s Indifference Curve')
plt.plot(portfolio_std_devs, utility_david, '--', label='David₩'s Indifference Curve')
plt.scatter(portfolio_std_devs[np.argmax(utility_john)], portfolio_returns[np.argmax(utility_john)], color='red', s=50,
label='John₩'s Optimal Portfolio')
plt.scatter(portfolio_std_devs[np.argmax(utility_david)], portfolio_returns[np.argmax(utility_david)], color='blue', s=50,
label='David₩'s Optimal Portfolio')
plt.title('Optimal Portfolios for Different Risk Aversions')
plt.xlabel('Risk (Standard Deviation)')
plt.ylabel('Expected Return')
plt.legend()
plt.grid(True)
plt.show()
```

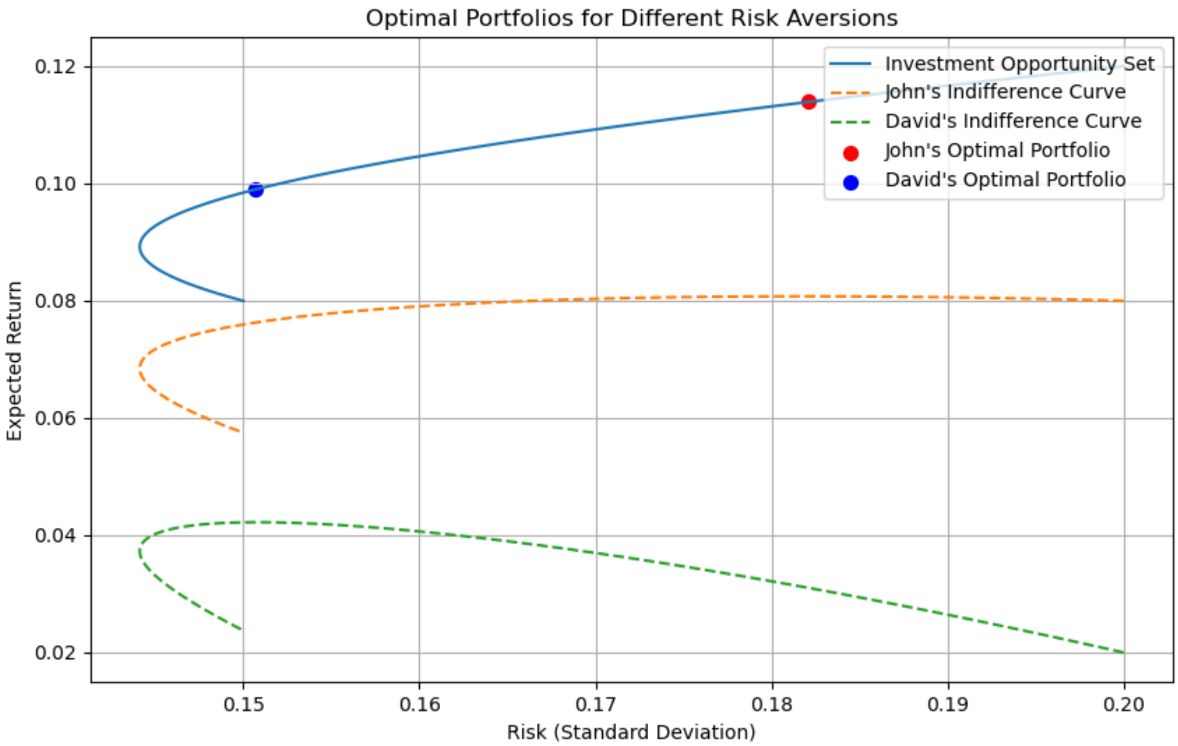
위의 코드에서 numpy.linspace 함수를 사용해 투자기회집합의 테이블에 주식 A, B에 각각 0에서 100까지의 가중치를 생성하였으며, 각 포트폴리오의 기대수익률과 표준편차를 수식으로 정의했습니다.

무차별 함수는 $U(x)=x-0.5\times y\times z^2$ 의 무차별곡선의 식을 사용하였으며, 각각 x는 포트폴리오의 기대수익률(returns)을, Y는 위험회피계수 (aversion)를, z는 포트폴리오의 표준편차(portfolio_std_devs)를 의미합니다.

* Y가 높을수록 위험회피가 높음

matplotlib.pyplot의 함수들을 사용해 그래프를 그립니다. plt.plot은 선을, plt.scatter은 점을 그립니다. np.argmax는 NumPy 라이브러리에서 제공하는 함수로, 주어진 배열에서 최대값을 가지는 요소의 인덱스를 반환합니다.
위의 코드의 결과는 아래와 같습니다. (Invest_Indifference Curve.py 참고)

※ 파이썬 실행창에 달성 가능한 투자기회집합 테이블의 전부가 표시됩니다.




아래는 파이썬 코드의 결과값입니다.

John's Optimal Portfolio:
  Weight in Stock A: 15.15%
  Weight in Stock B: 84.85%
  Expected Return: 11.39%
  Standard Deviation: 0.18
  Variance: 0.03

David's Optimal Portfolio:
  Weight in Stock A: 52.53%
  Weight in Stock B: 47.47%
  Expected Return: 9.90%
  Standard Deviation: 0.15
  Variance: 0.02

John 의 최적 포트폴리오는 A 주식에 15.15%, B 주식에 84.85%를 분배하여 표준편차(위험) 0.18 과 기대수익률 11.39%를 달성하는 것이고, David 의 최적 포트폴리오는 A 주식에 52.53%, B 주식에 47.47%를 분배하여 표준편차(위험) 0.15 와 수익률 9.90%를 달성하는 것입니다.

[ 해 석 ]

투자기회집합 곡선은 가능한 모든 투자 조합들 중에서 각각의 위험 수준에 대해 최대 수익률을 제공하는 포트폴리오들을 연결한 곡선입니다. 이 곡선은 일반적으로 포트폴리오의 위험(표준 편차)과 수익률을 축으로 하여 그려지며, 보통 위로 볼록한 형태를 가집니다. 이 형태는 주식의 수익률 분포가 정규분포를 따르고, 투자의 위험과 수익률이 서로 특정한 상관관계를 가지고 있음을 반영합니다.

무차별곡선은 투자자가 동일한 수준의 만족을 얻을 수 있는 포트폴리오의 위험과 수익률 조합을 연결한 곡선입니다. 이 곡선 역시 위로 볼록한 형태를 가지며, 이는 투자자가 추가적인 위험을 부담할 때 요구하는 추가 수익이 점차 증가한다는 것을 의미합니다. 즉, 위험을 더 많이 부담하기 위해서는 그만큼 더 많은 수익이 필요하다는 경제적 원칙을 반영합니다.

위험과 수익의 상관관계: 두 곡선 모두 포트폴리오의 위험과 수익 간의 상관관계를 기반으로 합니다. 투자기회집합은 가능한 최대 수익을 각 위험 수준에서 연결하는 반면, 무차별곡선은 같은 효용 수준을 유지하면서 위험과 수익의 조합을 연결합니다.

위로 볼록한 곡선: 두 곡선 모두 위로 볼록합니다. 이는 더 높은 위험을 감수할 경우, 그에 상응하는 더 높은 기대 수익률이 필요함을 나타냅니다. 투자기회집합에서는 효율적인 포트폴리오만이 표현되며, 무차별곡선에서는 투자자의 위험 회피 성향에 따라 최적의 포트폴리오가 결정됩니다.

최적성의 탐색: 두 곡선의 접점은 투자자에게 최적의 포트폴리오를 제공합니다. 이 점에서 투자자는 주어진 위험 수준에서 최대의 수익률을 달성하면서 자신의 효용을 최대화할 수 있습니다. 이 접점은 무차별곡선이 투자기회집합을 "접"하는 지점으로, 이는 수학적으로 그 투자자의 위험 선호와 시장에서 가능한 최고의 수익률 및 위험 조합이 일치하는 지점입니다.