KAIST MFE, 2024 Fall

Kim Hyeonghwan

2024-09-02

Table of contents

Welcome!

안녕하세요, KAIST MFE 24년 가을학기에 이수한 과목의 과제 등을 정리해두었습니다.

Part I

머신러닝('24 가을)

머신러닝 1주차

Part II

딥러닝('24 가을)

딥러닝 1주차

Part III

시뮬레이션 방법론('24 가을)

시뮬레이션방법론 1주차

블랙숄즈공식 예시

1. $f_t + \frac{1}{2}\sigma^2 S^2 f_{ss} + rSf_s - rf = 0$

-> 수치해석적인 방법으로 풀게 됨, FDM(Finite Difference Method)

2. $P(0) = e^{-rT}E^Q[P(T)]$

-> 마팅게일, 몬테카를로 시뮬레이션(Montecarlo simulation, MCS)을 주로 사용함

Volume과 적분

$x \sim uniform[0,1]$, $\alpha = E[f(x)] = \int_0^1 f(x)dx$

그러나, MCS를 이용하는 경우 임의변수 $x_1, x_2, ..., x_n$을 샘플링하여 $\hat{\alpha} = \frac{1}{N}\sum_i^N f(x_i)$로 산출함

두 값이 정확히 일치하지는 않지만, 표본이 커질수록 그 오차는 0으로 수렴함($\alpha \approx \hat{\alpha}$)

이는 **대수의 법칙과 중심극한정리**에 따라 수학적으로 정의할 수 있음

i 중심극한정리

표본평균($\hat{\alpha}$)은 정규분포를 따르므로, $\hat{\alpha} - \alpha \sim N(0, \frac{\sigma^2}{N})$
즉, 표본의 크기가 커질수록 두 차이는 0으로 수렴함(probibility convergence)
오차의 표준편차는 $\frac{\sigma}{\sqrt{N}}$이므로, 표본의 크기가 100배 증가하면 오차의 표준편차는 10배 감소함

이외에도 간단힌 사다리꼴(trapezoidal) 방식을 이용해볼 수 있음.

3. $\alpha \approx \frac{f(0)+f(1)}{2n} + \frac{1}{n}\sum_{i=1}^{n-1} f(\frac{i}{n})$ (   )

이는 매우 간단하고 효율적인 방법이지만, 변수가 늘어날 수록 효율이 급감함.

Part IV

이자율파생상품('24 가을)

12

이자율파생상품 1주차

Part V

수치해석학('24 가을)

수치해석학 1주차

강의 개요 : 금융수치해석의 필요성

주로 파생상품 평가와 최적화 방법론에 대해서 다룰 예정

파생상품 평가

$ds = rSdt + \sigma SdW^Q$

기하학적 브라운운동을 따르는 기초자산에 대한 파생상품의 가격 $f(t, S)$는 아래의 PDE로 표현됨

$f_t + \frac{1}{2}\sigma^2 S^2 f_{ss} + rSf_s - rf = 0$

이 블랙숄즈 미분방정식을 컴퓨터로 풀어내는 것이 주요 내용임

여기에는 반드시 연속적인 수식을 이산화하는 과정이 필요하며, 다양한 수치해석적인 기법이 활용됨

대표적으로 유한차분법(Finite Difference Method, FDM)이 존재

최적화 방법론

이외의 다양한 최적화방법론은 시간이 여유롭다면 이것저것 다룰 예정

- Minimum Variance Portfolio : Single-period에 대해 Sharpe ratio 극대화 등
- Stochastic programming : Multi-period에 대해 Minimum var 문제 해결 등
- Non-convex optimization : 미분을 통해 극값을 산출할 수 없는 경우의 최적화
- Parameter estimation 또는 Model calibration : $min_{\theta,\sigma,k} \sum(model\ price - market\ price)^2$ 와 같은 문제 등

컴퓨터 연산에 대한 이해

수치해석기법을 사용할 때 필연적으로 오차(error) 발생

1. Truncation error : 연속적인 수학적인 모델을 이산화하면서 발생하는 오차(e.g. 미분계수)
2. Rounding error : 컴퓨터 시스템상 실수(real number)를 정확히 표현할 수 없는 데에서 기인(2진법 vs. 10진법)

```
import numpy as np

a = 0.1

print(a+a+a==0.3,a+a+a+a==0.4)
```

```
False True
```

Part VI

금융시장 리스크관리('24 가을)

금융시장 리스크관리 1주차

Part VII

미시경제학('24 가을)

미시경제학 1주차

Part VIII

글로벌 지속가능회계('24 가을)

글로벌 지속가능회계 1주차