Black-Scholes Implied Volatility for SPX

최현태

10th June 2021

Abstract

효율적 시장가설에 의하면 현재 주가는 시장에서 가용할 수 있는 모든 정보를 내포하고 있기 때문에 과거의 주가를 토대로 한 historical volatility(주가 수익률의 표준편차)는 미래의 주가 예측에 별 다른 도움을 주지 못한다. 내재변동성(Implied Volatility)는 주식(혹은 주가, 원자산)이나 파생상품을 거래할 때 매우 유용한 지표이다. 내재변동성은 첫째 원자산의 변동성에 대한 하나의 추정치(estimation)으로서 원자산의 출렁임이 어느 정도일지 가늠할 수 있게 해준다. 그 결과, 둘째 원자산 가격의 확률분포를 예상할 수 있게 해준다. 내재변동성을 구하기 위해서는 옵션에 대한 정보가 필요하다. 구체적으로 옵션 가격, 현재시점, 행사시점, 행사가격, 원자산가격, 무위험이자율이 있으면 Black-Scholes 방정식을 사용하여 수치적으로 해당 옵션 정보에 대응하는 ImVol을 구할 수 있다.

이론적으로는 옵션의 행사가격과 행사시점을 기준으로 연속적인 3차원 내재변동성 곡면을 생각해볼 수 있다. 다만 현실에서 얻을 수 있는

옵션에 대한 정보는 상당히 제한적이다. 우선 행사가격, 행사시점 등이 이산적으로 주어질 뿐만 아니라 행사가격과 행사시점이 매우 제한된 범위 내에서만 주어진다. 이와 같은 현실적 이유로 인해 curve fitting 또는 surface fitting을 통해 주어진 옵션 정보에 대해 구한 ImVol을 이용하여 내재변동성 곡선와 내재변동성 곡면을 추정할 수 있다. 여기서 내재변동성 곡선는 2차원 곡선으로 내재변동성을 행사시점에 대해 구한 것이고, 내재변동성 곡면은 3차원 곡면으로 내재변동성을 행사시점과 행사가격에 대해 구한 것이다.

이 보고서에서는 S&P 500 Index(SPX)의 옵션에 대한 정보를 가지고 수치적 방법을 통해 ImVol을 계산한 다음 이 정보들을 이용하여 내재변동성 곡선과 내재변동성 곡면을 추정해본다.

이 보고서는 S&P 500 Index(SPX) 옵션 가격을 바탕으로 내재변동성 (implied volatility)을 계산한 뒤 다항회귀(polynomial regression)과 스플라인(spline), radial basis function(RBF) interpolation을 통해 내재변동성 곡선(행사가격-내재변동성)과 내재변동성 곡면(만기-행사가격-내재변동성)을 추정해본다. 구체적인 데이터와 추정방법에 대해 논의하기 전에 이론적으로 내재변동성을 계산하기 위한 블랙-숄즈식(Black-Scholes Formula)의 도출과정을 간략히 요약한 다음 내재변동성의 개념, 블랙-숄즈식으로부터 내재변동성을 구하기 위한 수치적 방법인 뉴턴-랩슨 방법, 옵션 만기의 원자산 가격(혹은 원자산)의 위치를 나타내는 forward moneyness의 개념에 대해 소개한다. 이후 SPX 옵션 데이터를 바탕으로 내재변동성을 계산한 다음 행사가격에 대해 다항회귀를 통해 내재변동성 곡선을 추정해보고 마지막으로 만기와 forward moneyness에 대해 스플라인, ratial basis function interpolation을 통해 내재변동성 곡면을 추정한다.

1 이론적 배경: 블랙-숄즈 모형과 내재변동성

블랙-숄즈(Black-Scholes) 모형 혹은 블랙-숄즈-머튼 모형은 파생상품을 포함하는 동태적 금융시장에 관한 수리적인 모형이다. 구체적으로 블랙-숄즈 모형에서는 블랙-숄즈 방정식이라고 불리는 확률편미분방정식(stochastic partial differential equation, SPD, 또는 SPDE)으로부터 블랙-숄즈 식이라고 불리는 유러피안 옵션의 가격 추정량을 도출할 수 있다. 흔히 블랙-숄즈식은 원자산의 위험과 기대수익이 주어져 있을 때 블랙-숄즈 방정식으로 구성되는 경계값문제(boundary value problem)의 유일한 해라고 평가된다. 그러나 여기에 대해서는 Choi-Choi(2018)에서 경계값 문제의 해가 무한히 많은 것으로 밝혀졌으며 따라서 블랙-숄즈 식이 경제학의 근본 원리인 일물일가의 법칙에 위배된다고 할 수 있다.[6](이 보고서에서는 블랙-숄즈 방정식으로부터 unique한 블랙-숄즈 식이 도출되지 않는다는 논의는 더 이상 이어가지 않는다. 이에 대한 자세한 논의는 CC(2018)[6]와 KKCC(2020)[7]를 참조할 수 있다.)

블랙-숄즈 모형 뒤에 있는 핵심적인 아이디어는 델타 헷징(delta hedging)이라고 불리는 옵션 전략을 통해 원자산과 관련된 위험을 제거하는 것이다. 델타 헷징은 수식에 의해 계산된 "옳은" 방식대로 옵션을 사고 팖으로써 원자산과 관련된 위험을 제거하는 옵션전략으로 특히 블랙-숄즈 모형에서는 연속 수정 델타-헷징(continuously revised delta hedging)이라는 전략을 사용한다.

블랙-숄즈 모형은 옵션 시장 참가자들에 의한 조정을 거쳐 매우 폭넓게 쓰이고 있다. 모형의 가정에 대해 여러 각도로 일반화하려는 시도가 있었으며 그 결과 블랙-숄즈 모형은 파생상품 가격 결정과 위험 관리에 사용되는 하나의 모형군을 이루게 되었다. 블랙-숄즈 식에는 시장에서 관찰할 수 없는 하나의 퍼러미터가 존재한다. 그것은 원자산의 수익률(returns)의 표준편차를 나타내는 volatility이다. 이 때 volatility는 과거의 원자산 수익률의 표준편차인 historical volatility와는 다르다.

효율적 시장가설에 의하면 현재 주가는 시장에서 가용할 수 있는 모든 정보를 내포하고 있기 때문에 과거의 주가를 토대로 한 historical volatility(주가 수익률의 표준편차)는 미래의 주가 예측에 별 다른 도움을 주지 못한다. 따라서 유러피언 옵션의 공정한 가격을 평가하기 위해서는 시장이 판단하는 미래의 volatility를 구해야 한다. 여기서 volatility를 추정하는 방식에는 여러 가지가 있을 수 있지만 대표적으로 다른 옵션 가격을 주어진 것으로 보고 volatility를 계산하는 것이다. 블랙-숄즈 식에서 옵션 가격(Put이든 Call이든)은 volatility에 대해 정(+)의 관계를 가지기 때문에 역으로 옵션 가격을 통해 volatility를 만들어 내는 것이 가능하고(내재변동성) 이렇게 구한 implied volatilty는 다른 모형을 calibrate하는 데에 사용된다. 이렇게 구한 volatility를 내재변동성라고 한다.

1.1 Black-Scholes Environment

블랙-숄즈 방정식은 블랙-숄즈 환경(Black-Scholes Environment)에서 성립한다. 블랙-숄즈 환경에 대해 알아보는 것의 의의는 블랙-숄즈식을 거쳐 구한 내재변동성의 함의를 정확히 이해하고 그 한계에 대해 사전에 인지함으로써 내재변동성에 대한 맹신과 맹목적인 사용을 방지하는 것이다. 더 나아가 블랙-숄즈 환경에 대해 정확히 이해해야만 블랙-숄즈식을 통해 도출한 내재변동성을 현실에 제대로 적용할 수 있다. 블랙-숄즈 환경은 다음과 같은 여덟 가지 가정으로 구성된다.

1. 최소한 하나의 위험 자산과 하나의 무위험자산이 존재한다.
2. 무재정조건(No arbitrage condition)이 성립한다.
3. 무한한 유동성 조건 1: 무위험자산을 무위험이자율에 원하는만큼 빌리거나 살 수 있다.

4. 무한한 유동성 조건 2: 공매도를 포함하여위험 자산을 원하는만큼 빌리거나 살 수 있다.

5. 마찰이 존재하지 않는 시장: 수수료 등 거래 비용이 존재하지 않는다.

6.무위험이자율은 상수이다.

7. 원자산은 기하 브라운 운동(Geometric Brownian Motion)을 따른다.

8. drift μ와 volatility σ는 상수이다.

1.2 Black-Scholes Stochastic Partial Differential Equation

먼저 블랙-숄즈 환경에서 원자산 S_t는 다음과 같은확률편미분방정식을 따른다.

$$dS = \mu(S_t, t)dt + \sigma(S_t, t)dW_t^P \tag{BS1}$$

여기서 $\{dW_t^P\}$는 서로 독립이며 평균이 0이고 분산이 dt인 정규분포를 따르는 확률과정이다. $\mu_t = \mu(s_t, t)$ 그리고 $\sigma_t = \sigma(S_t, t)$라고 표기하고 파생상품의 가치를 $F(S_t, t)$라고 할 때 포트폴리오 전체의 가치는

$$P_t = \theta_{1,t} F(S_t, t) + \theta_{2,t} S_t \tag{BS2}$$

와 같이 나타난다. Ito-Doeblin lemma에 따르면

$$dP_t = \theta_{1,t} dF(S_t, t) + \theta_{2,t} dS_t \tag{BS3}$$

$$dF_t = [F_t + \mu_t F_S + \frac{1}{2}\sigma_t^2 F_{SS}]dt + \sigma_t F_S dW^P \tag{BS4}$$

가 성립한다. (BS3)과 (BS4)를 더하면

$$dP_t = \theta_{1,t}[F_t + \frac{1}{2}F_{SS}\sigma_t^2]d_t + [\theta_{1,t}F_S + \theta_{2,t}]dS_t \tag{BS5}$$

를 도출할 수 있다. 델타 헷징 전략은 아래와 같이 나타난다.

$$\theta_{1,t} = 1, \theta_{2,t} = -F_S \tag{BS5}$$

이제 (BS5)와 (BS5)를 결합하면

$$dP_t = \theta_{1,t}[F_t + \frac{1}{2}F_{SS}\sigma_t^2]dt \tag{BS7}$$

$$dP_t = rP_t dt \tag{BS8}$$

이 도출되는데 마지막으로 (BS1), (BS7), 그리고 (BS8)을 결합하면 블랙-숄즈 확률편미분방정식

$$-rF + rF_S S_t + F_t + \frac{1}{2}S_t^2 F_{SS}\sigma_t^2 = 0 \tag{BS9}$$

이 나타난다.

이제 만기시점 T에서 이 파생상품의 지불금액함수가 다음과 같다고 하자.

$$F(S_T, T) = G(S_T, T) \tag{TC}$$

여기서 G는 주어진 함수이다. 예를 들어 행사가격이 K인 유럽형콜옵션과 풋옵션의 지불금액함수는 다음과 같다.

$$F(S_T, T) = [S_T - K]^+ \tag{Call Payoff}$$

$$F(S_T, T) = [K - S_T]^+ \tag{Put Payoff}$$

여기서 $A^+ = maxA, 0$이다. 이제 확률편미분방정식 (BS9)과 이에 해당하는 말기조건 (TC)으로 구성된 경계값 문제(boundary value problem)을 풀어서 파생상품가지 $F(S_t, t)$를 구할 수 있다.

1.3 Black-Scholes Stochastic Formula

(BS9)를 약간만 수정하면 아래와 같이 바뀔 수 있다.

$$-rF + r\frac{1}{\frac{1}{S_t}}F_S + F_t - \frac{1}{2}\frac{1}{-\frac{1}{S_t^2}}F_{SS}\sigma_t^2 = 0 \tag{BS10}$$

그리고 최종적으로

$$-rF + r\frac{\partial F}{\partial \log S_t} + F_t - \frac{1}{2}\frac{\partial^2 F}{(\partial \log S_t)^2}\sigma_t^2 = 0 \tag{BS11}$$

와 같이 나타낼 수 있다. 그러면 블랙-숄즈 편미분방정식은 다름 아닌 열전도방정식이 되고 이에 해당하는 콜옵션과 풋옵션의 말기조건 (Call Payoff), (Put Payoff)을 고려하여 경계값문제를 풀 수 있다. 그 결과로 나타나는 블랙-숄즈식은 유럽형 콜옵션과 유럽형 풋옵션의 시점 t에서 공정한 가치를 다음과 같이 나타낸다.

$$C_t^{BS} = S_tN(d_1) - Ke^{-rt}N(d_2) \tag{BS12}$$

$$P_t^{BS} = Ke^{-rt}N(-d_2) - S_tN(-d_1) \tag{BS13}$$

여기서 $N(z)$, d_1과 d_2는 각각 다음과 같다.

$$N(z) = \int_{-\infty}^{z} \frac{1}{\sqrt{2\pi}} \exp(-\frac{1}{2}x^2)dx \tag{BS14}$$

$$d_1 = \frac{1}{\sigma\sqrt{\tau}}(\log(\frac{S_t}{K} + [r + \frac{\sigma^2}{2}]r), d_2 = d_1 - \sigma\sqrt{\tau} \tag{BS14}$$

1.4 Implied Volatility

과거의 원자산 시계열 자료(time series data)을 바탕으로 추정된 변동성을 역사적 변동성(historical volatility)라고 한다. 어떤 금융상품의 제i번째 거래일의 종가를 S_i라고 하면, 제i번째 거래일의 로그수익률은 $R_t = \log\frac{S_i}{S_{i-1}}$이다.

이 로그 수익률을 일차수익률이라고 부른다. 블랙-숄즈 환경에서는 $\{R_i\}$의 표준편차를 역사적 변동성이라고 할 수 있다. 즉 과거 m일간 데이터를 사용하여 추정한 역사적 변동성을 m일 역사적 변동성이라고 부른다. 변동성을 연단위로 표현하는 것이 일반적이므로 일차수익률에 1년 간 영업일수 252를 기준으로 하여 일차수익률의 표준편차에 $\sqrt{252}$를 곱한 값이 연간 변동성(annual volatility)로 이용된다. 그러나 효율적 시장가설에 의하면 현재 주가는 시장에서 가용할 수 있는 모든 정보를 내포하고 있기 때문에 과거의 주가를 토대로 한 historical volatility(주가 수익률의 표준편차)는 미래의 주가 예측에 별 다른 도움을 주지 못한다.

앞에서 언급한 것처럼 블랙-숄즈식((BS12) - (BS14))

$$C_t^{BS} = S_t N(d_1) - Ke^{-rt}N(d_2) \tag{BS12}$$

$$P_t^{BS} = Ke^{-rt}N(-d_2) - S_t N(-d_1) \tag{BS13}$$

$$N(z) = \int_{-\infty}^{z} \frac{1}{\sqrt{2\pi}} \exp(-\frac{1}{2}x^2)dx \tag{BS14}$$

$$d_1 = \frac{1}{\sigma\sqrt{\tau}}(\log(\frac{S_t}{K} + [r + \frac{\sigma^2}{2}]r), d_2 = d_1 - \sigma\sqrt{\tau} \tag{BS14}$$

에서 원자산 가격은 상수인 변동성 σ에 대해 기하 브라운 운동(GBM)을 따르며 BS 모형에 따라 주어진 원자산가격에 대하여 유러피안 옵션의 공정한 가치가 유일하게 존재한다. 이 때 옵션 가격은 원자산의 변동성에 대해 단조증가한다. 높은 변동성은 해당 유러피언 옵션이 ITM(in the money)일 확률을 증가시키기 때문에 유러피언 옵션의 가격 역시 비싸진다. 여기서 ITM은 만기일에 원자산이 유러피언 옵션의 내재 가치(intrinsic value)가 양이 되는 가격을 지닌다는 의미이다. moneyness의 개념은 다음 절에서 소개한다.

이제 블랙-숄즈식에 따라 구한 풋옵션과 콜옵션의 가격을 각각 $C^{BS}(t, S; K, T, \sigma, r)$ 과 $P^{BS}(t, S; K, T, \sigma, r)$ 라고 할 때 각 옵션의 시장 가격을 이용하면 내재변동성 σ^{BS}_{imp}를 구할 수 있다. 콜 옵션의 경우 C^{mkt}을 옵션의 시장가격이라고 하면

$$C^{BS}(t, S; K, T, \sigma^{BS}_{imp}, r) = C^{mkt}(K, T)$$

를 만족한다. 즉 내재변동성 σ^{BS}_{imp}는 블랙-숄즈식에서 현재 시점의 옵션 가격을 만들어내는 변동성이다. 내재변동성는 가치평가식에 따라 Heston 내재변동성, SABR 내재변동성 등이 존재하기도 하지만 이 보고서에서는 블랙-숄즈식에 따라 평가된 내재변동성만 논의한다. 따라서 σ^{BS}_{imp}를 앞으로는 σ_{imp}로 표기한다.

내재변동성를 계산하는 것은 내재변동성 방정식 $f(\sigma_{imp}) = 0$,

$$f(\sigma_{imp}) = C^{mkt}(K, T) - C^{BS}(t, S_t; K, T, \sigma_{imp}, r)$$

을 푸는 것과 동치이다. 여기서 내재변동성 방정식은 비선형방정식이므로 해석해(closed-form solution)가 존재하지 않는다. 따라서 수치적인 방법으로 σ_{imp}를 구해야 한다. 즉, 내재변동성 방정식 $f(\sigma_{imp}) = 0$을 만족하는 σ_{imp} 를 구하는 대신 $f(\sigma_{imp})$를 매우 작게 만드는 σ_{imp}를 구하는 것이다. 이와 같이 수치적인 방법으로 비선형방정식의 해를 근사할 때 나타나는 문제점은 initial condition을 어떻게 설정하느냐에 따라서 근사값이 크게 달라질 수도 있다는 것이다. 뿐만 아니라 initial condition에 따라서 σ_{imp}가 수렴하지 않을 수도 있다. 수치적인 방법을 사용하여 근사해를 구할 때 초기값 설정은 매우 중요한 문제이다. 수치적인 방법을 사용하여 최적 퍼러미터의 근사해를 구하는 딥러닝에서도 퍼러미터 의 초기값 설정(딥러닝에서는 주로 weight initialization 이라고 명명한다)은 매우 중요한 문제이다. 이에 대한 자세한 논의는 점화식 연구[10]와 Improving The Way Neural Networks Learn[12]에서 찾을 수 있다.

1.5 Newton-Raphson Method

이 보고서에서는 비선형방정식의 해를 구하기 위해 뉴턴-랩슨(Newton-Raphson) 방법을 사용한다. 연속이고 미분가능한 함수 $f(x) = 0$ 의 해를 구할 때 해가 $x = x_0$ 근처에 있다는 정보를 알고 있다고 하자. 이 경우 뉴턴-랩슨 방법에 따르면 더 좋은 근사해는

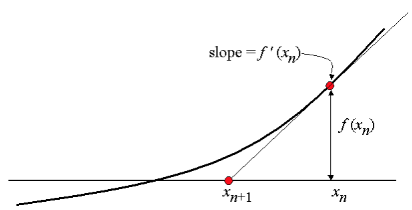
$$x_1 = x_0 - \frac{f_0}{f'(x_0)}$$

라고 할 수 있다. 이 과정은 해가 원하는 정확도에 들어올 때까지

$$x_{n+1} = x_n - \frac{f_n}{f'(x_n)}$$

을 반복할 수 있다. 이 방법에 대한 간단한 시각적인 설명을 위해 아래 그래프를 제시할 수 있다.

Figure 1: Newton-Raphson Method



위의 그래프에는 $x = x_n$에서 f에 대한 접선이 나타나 있다. 이 접선의 기울기는 $f'(x_n)$이고 $(x_n, f(x_n))$을 지나므로 이 접선의 방정식은 $g(x) = f'(x_n)(x - x_n) + f(x_n)$이다. 이제 이 방정식의 해를 구하면 $g(x) = 0$에서 $f'(x_n)(x - x_n) + f(x_n)$이므로 $x = x_n - \frac{f_n}{f'(x_n)}$이 도출된다. 이 x는 $f = 0$에 대한 더 나은 근사해라고 할 수 있으므로 $x = x_{n+1}$로 놓으면 앞서 확인한 점화식을 얻을 수 있다.

1.6 Moneyness

moneyness의 개념에 대해 위키피디아(Wikipedia)의 설명을 인용하면[8]

> In finance, moneyness is the relative position of the current price (or future price) of an underlying asset (e.g., a stock) with respect to the strike price of a derivative, most commonly a call option or a put option. Moneyness is firstly a three-fold classification: if the derivative would have positive intrinsic value if it were to expire today, it is said to be in the money; if it would be worthless if expiring with the underlying at its current price it is said to be out of the money, and if the current underlying price and strike price are equal, it is said to be at the money. There are two slightly different definitions, according to whether one uses the current price (spot) or future price (forward), specified as "at the money spot" or "at the money forward", etc.
>
> This rough classification can be quantified by various definitions to express the moneyness as a number, measuring how far the asset is in the money or out of the money with respect to the strike – or conversely how far a strike is in or out of the money with respect to the spot (or forward) price of the asset. This quantified notion of moneyness is most importantly used in defining the relative volatility surface: the implied volatility in terms of moneyness, rather than absolute price. The most basic of these measures is simple moneyness, which is the ratio of spot (or forward) to strike, or the reciprocal, depending on convention.

와 같다. 즉 moneyness는 원자산의 현재 혹은 미래 가격의 행사가격(K)에 대한 상대적인 위치를 나타내는 표현으로 ITM, ATM, 그리고 OTM 의 세 가지로 나뉜다. 인용한 위키피디아 내용의 마지막 부분에 언급되어 있는 것처럼 moneyness는 다양한 방식으로 측정될 수 있는데 이 보고서에서는 대표적으로 세 가지 방법을 소개한다.[11]

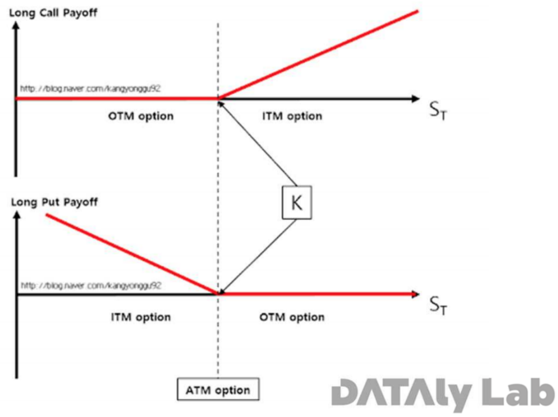
1. simple moneyness: $\frac{S}{K}$
2. log moneyness: $\log \frac{S}{K}$
3. standardized log moneyness: $\frac{\log SK}{\sigma\sqrt{\tau}}$

τ는 현재 시점(t)과 만기(T)의 차이를 나타낸는 변수이다. 이 세 가지 방법은 현재가격을 기준으로 moneyness를 표현한 것이다. 만약 forward price F를 기준으로 moneyness를 나타낼 경우 moneyness는

1. simple moneyness: $\frac{F}{K}$
2. log moneyness: $\log \frac{F}{K}$
3. standardized log moneyness: $\frac{\log FK}{\sigma\sqrt{\tau}}$

와 같이 나타나며 forward moneyness라고 불린다. 현재가격(S)이 주어져 있을 때 forward price F는 연속복리 가정 하에서 무위험이자율을 통해 $F = Se^{r\tau}$와 같이 계산할 수 있다. 이와 같은 forward moneyness는 블랙-숄즈식이 작동하는 방식과 부합한다고 할 수 있다. 유러피언 옵션에서 롱포지션을 취할 때 (콜옵션과 풋옵션을 매수할 때) moneyness를 그래프로 나타내면 아래와 같다.[9]

Figure 2: Taking long position for European options and moneyness



2 SPX 내재변동성 계산

2.1 데이터

SPX 옵션 데이터는 barchart 웹사이트에서 조건을 설정하여 다운로드하였다. 자료수집일인 5월 22일을 기준으로 대략 한 달을 기준으로 6월 21일, 7월 30일, 9월 17일이 만기인 콜옵션과 풋옵션 데이터를 수집하였다. 이 중 변동성 곡선 추정에는 6월 21일 데이터를 사용하였으며 변동성 곡면 추정에는 세 가지 만기의 데이터를 모두 사용하였다. 6월 21일이 만기인 원데이터를 python의 모듈 pandas로 import한 뒤에 간략하게 살펴보면 다음과 같다.

Figure 3: SPX European Options Raw Data(head)

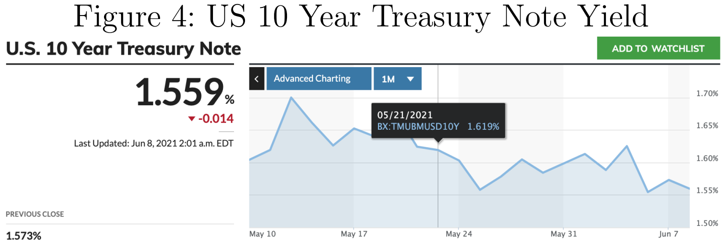
	Strike	Moneyness	Bid	Midpoint	Ask	Last	Change	%Chg	Volume	Open Int	Vol/OI	IV	Type	Last Trade
0	4,075.00	+1.95%	131.8	132.6	133.4	135.42	135.42	unch	53.0	0.0	0.0	18.70%	Call	05/21/21
1	4,080.00	+1.83%	127.9	128.7	129.5	128.43	128.43	unch	6.0	0.0	0.0	18.46%	Call	05/21/21
2	4,090.00	+1.58%	120.1	120.9	121.7	139.85	139.85	unch	4.0	0.0	0.0	18.14%	Call	05/21/21
3	4,100.00	+1.34%	112.6	113.4	114.2	115.44	115.44	unch	87.0	0.0	0.0	17.80%	Call	05/21/21
4	4,110.00	+1.10%	105.1	105.9	106.7	108.38	108.38	unch	10.0	0.0	0.0	17.43%	Call	05/21/21

데이터를 살펴보면 먼저 Strike(행사가격), Bid(매수자의 입찰가격), Ask(매도자의 입찰가격), Volume(거래량)이 눈에 들어온다. 뿐만 아니라 앞에서 소

개한 Moneyness(현재가격 기준), IV(내재변동성)까지 제시해 주는 것을 알 수 있다. Moneyness와 내재변동성은 오픈 데이터 소스에서 하나의 변수로 제공될만큼 금융데이터 분석에 있어서 기본적이면서도 중요한 변수이다. 그러나 이 보고서에서는 이 데이터들을 직접 사용하지 않고 이전의 논의를 바탕으로 직접 계산해본다. 계산 방식이 다르기 때문에 원데이터에서 제공한 변수값과 이 보고서에서 계산된 값은 일치하지 않는다. Moneyness 변수의 경우 Strike에 대한 5월 22일의 SPX 값 4155.86의 상대적인 위치를 나타낸 것으로 forward moneyness가 아니다. 또한 내재변동성의 경우 초기값 어떻게 설정하느냐, 모수에 어떤 값을 대입하느냐에 따라 차이가 날 수 있다.

이 보고서에서는 무위험이자율을 $(\frac{\text{US 10 Year Treasury Note Yield}}{\text{FOMC Target Inflation Rate}} - 1) \times 100$으로 설정하였다. 즉 명목이자율을 미재무성 10년 국채 수익률로, 물가상승률을 미연방공개시장위원회(Federal Open Market Committee, FOMC)가 설정한 목표물가상승률로 지정하였다. 데이터수집일인 5월 22일을 기준으로 미재무성 10년 국채 수익률은 약 1.6%이다. 또한 평균물가목표제(average inflation targeting)[15] 하에서 미국의 물가상승률은 장기적으로 2%에 수렴할 것으로 예상된다. 이렇게 계산된 무위험이자율은 약 3.9%이다. 또한 만기까지의 기간(τ)은 만기까지의 일수를 365로 나누어 계산된다. 금융데이터에서 일자를 계산하는 다양한 방식이 존재한다. 일자 계산 방식에 대한 자세한 논의는 금융공학 III: Introduction to Financial Engineering[4]에서 확인할 수 있다.

Figure 4: US 10 Year Treasury Note Yield



앞에서 논의한 블랙-숄즈식과 뉴턴-랩슨 방법에 따라 파이썬 코드를 작성하

면 Listing 1와 같다. Listing 1에서 BS_call은 콜옵션 가격을 계산하는 블랙-숄즈식이고 BS_put은 풋옵션 가격을 계산하는 블랙숄즈식이다. Newton_Raphson_sigma_call은 콜옵션 시장가격을 토대로 내재변동성을 계산하는데 그 과정에서 BS_vega 함수를 call한다.

```python
import numpy as np
from scipy.stats import norm

def BS_call(t, S, K, T, r, sigma):
    tau = T-t
    d1 = (np.log(S/K) + (r + 0.5*sigma**2)*tau) / (sigma*np.sqrt(tau))
    d2 = d1 - sigma * np.sqrt(tau)
    return S * norm.cdf(d1) - np.exp(-r * tau) * K * norm.cdf(d2)

def BS_put(t, S, K, T, r, sigma):
    tau = T-t
    d1 = (np.log(S/K) + (r + 0.5*sigma**2)*tau) / (sigma*np.sqrt(tau))
    d2 = d1 - sigma * np.sqrt(tau)
    return np.exp(-r * tau) * K * norm.cdf(-d2) - S * norm.cdf(-d1)

def BS_vega(t, S, K, T, r, sigma):
    tau = T - t
    d1 = (np.log(S / K) + (r + 0.5 * sigma ** 2) * tau) / (sigma * np.sqrt(tau))
    return S * norm.pdf(d1) * np.sqrt(tau)

```

```python
def NewtonRaphson_sigma_call(target_value, t, S, K, T, r, *args):
    max_iter = 1000
    precision = 1.0e-8
    sigma = 0.1
    for i in range(max_iter):
        price = BS_call(t, S, K, T, r, sigma)
        vega = BS_vega(t, S, K, T, r, sigma)
        diff = target_value - price
        if (abs(diff) < precision):
            return sigma
        sigma = sigma + diff/vega
    return sigma

def NewtonRaphson_sigma_put(target_value, t, S, K, T, r, *args):
    max_iter = 1000
    precision = 1.0e-8
    sigma = 0.2
    for i in range(max_iter):
        price = BS_put(t, S, K, T, r, sigma)
        vega = BS_vega(t, S, K, T, r, sigma)
        diff = target_value - price
        if (abs(diff) < precision):
            return sigma
        sigma = sigma + diff/vega
    return sigma
```

Listing 1: Newton-Raphson Methods Black-Scholes Implied Volatility

Figure 5 콜옵션 데이터를 변동성 곡선 추정을 위해 조정한 것이다.

Figure 5: SPX Call Option Data For Implied Volatility Curve

	Midpoint	t	St	Strike	T	r	Ivol
0	132.6	0.386301	4155.86	4075.0	0.471233	0.039216	0.159710
1	128.7	0.386301	4155.86	4080.0	0.471233	0.039216	0.158471
2	120.9	0.386301	4155.86	4090.0	0.471233	0.039216	0.155754
3	113.4	0.386301	4155.86	4100.0	0.471233	0.039216	0.153386
4	105.9	0.386301	4155.86	4110.0	0.471233	0.039216	0.150666

콜옵션의 가격은 Bid와 Ask의 평균인 Midpoint를 사용한다. 또한 t는 만기까지의 기간을 나타내는 변수이며 T는 만기, S_t는 5월 22일 SPX, $Ivol$은 뉴턴-랩슨 방법을 통해 계산된 내재변동성이다. 내재변동성을 %로 나타낸 다음 행사가격(K)에 대해 산포도를 그리면 Fiture 6와 같다.

Figure 6: SPX Call Option Data For Implied Volatility Curve

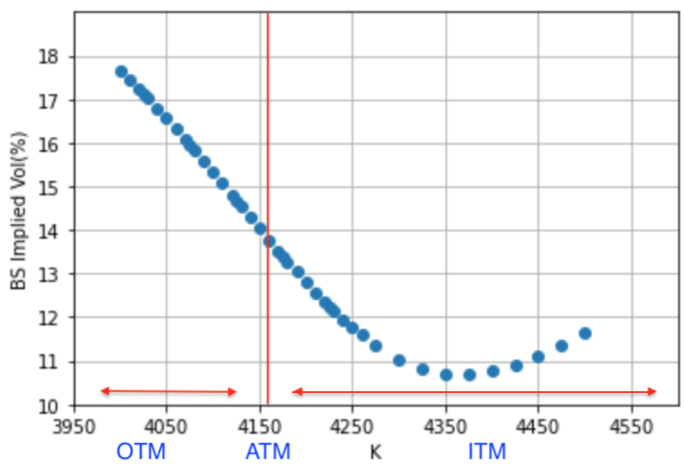


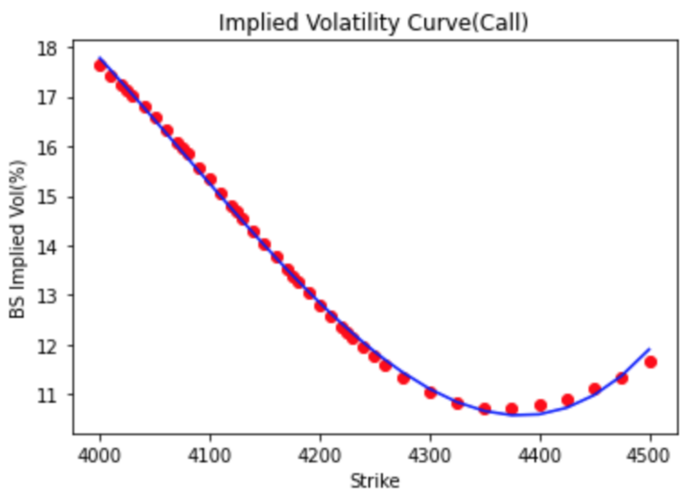
Figure 6는 원데이터에서 행사가격이 4000 이상 4500 이하인 콜옵션을 대상으로 하였다. Figure 6에서 사람이 웃고 있는 듯한 그래프를 확인할 수 있다. 이와 같은 내재변동성 곡선을 내재변동성 Smile이라고 한다. 내재변동성 Smile는 ATM에서의 내재변동성이 ITM에서의 내재변동성보다 낮게 나타나는 현상을 가리킨다. 변동성 Smile는 블랙-숄즈 모형이 변동성을 상수로 가정하고

원자산의 수익률이 log-normal 분포를 따른다고 가정하는 데에서 기인하는 문제라고 파악할 수 있다. 그러나 이 보고서에서는 다루지 않는 확률적 변동성 모형(대표적으로 local volatility model)에서도 나타난다. 변동성 Smile은 ATM 옵션보다는 ITM이나 OTM 옵션의 수요가 더 높은 현상을 나타낸다.

3 SPX 내재변동성 Curve 추정

앞의 산포도를 자료에 다항회귀분석을 실시하면 통해 변동성 곡선을 추정할 수 있다. 이 보고서에서는 4차 다항회귀를 위해 Listing 2와 같이 파이썬 코드를 작성하였다. 이렇게 회귀분석한 결과는 Figure 7에 나타난다. 회귀분석에 대해서는 회귀분석 상, 하[1]를 참조하였다.

Figure 7: Implied Volatility Curve for SPX




```
# Fitting Polynomial Regression to the dataset
from sklearn.preprocessing import PolynomialFeatures
from sklearn.linear_model import LinearRegression

```

```python
poly_reg = PolynomialFeatures(degree=4)
X = dfCall['Strike'].values.reshape(-1, 1)
y = (dfCall['Ivol'] * 100).values.reshape(-1, 1)
X_poly = poly_reg.fit_transform(X)
pol_reg = LinearRegression()
pol_reg.fit(X_poly, y)

# Visualizing the Polymonial Regression results
def viz_polymonial():
    plt.scatter(X, y, color='red')
    plt.plot(X, pol_reg.predict(poly_reg.fit_transform(X)),
   color='blue')
    plt.title('Implied Volatility Curve(Call)')
    plt.xlabel('Strike')
    plt.ylabel('BS Implied Vol(%)')
    plt.show()
    return
viz_polymonial()
```

Listing 2: Polynomial Regression

이렇게 구한 내재변동성 곡선을 다시블랙-숄즈식에 대입하면 관측되지 않은 콜옵션가격을 추정할 수 있다. 이와 같은 옵션 가격 추정은 내재변동성 곡선 추정이 왜 유용한지를 드러낸다. 추정을 위해서 Listing 3 과 같이 파이썬 코드를 작성했다.

```python
dfCall['est.'] = BS_call(dfCall['t'], dfCall['St'], dfCall['
   Strike'], dfCall['T'], dfCall['r'], \
pol_reg.predict(X_poly).reshape(-1)/100)
```

Listing 3: Newton-Raphson Methods Black-Scholes Implied Volatility

동일한 분석을 SPX 풋옵션에 대해서 수행한 결과는 아래와 같다.

Figure 8: Est. Call Option Price Curve

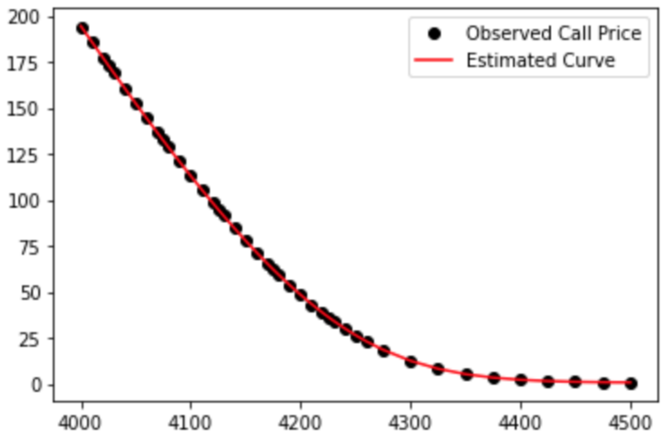


Figure 9: Scatter Plot(Put)

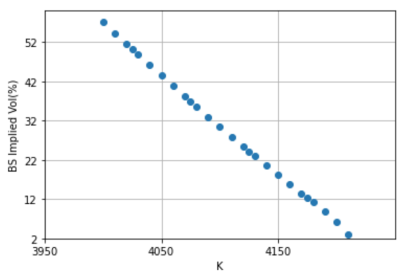
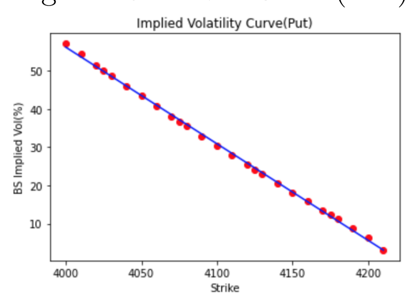


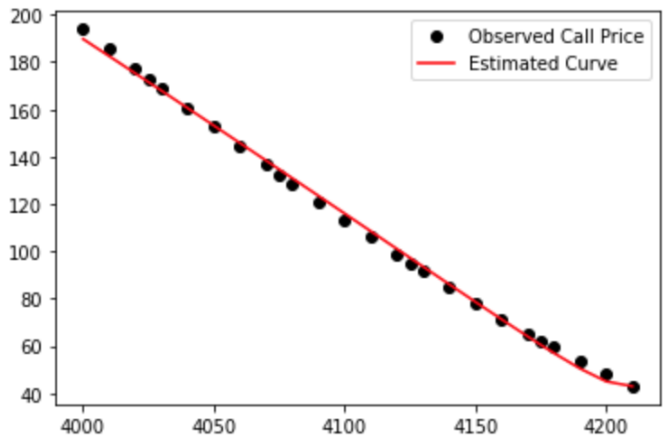
Figure 10: ImVol Curve(Put)



4 SPX 내재변동성 Surface 추정: Spline과 RBF

앞에서 수집한 7월 30일과 9월 17일이 만기인 SPX 옵션데이터를 함께 보면 만기까지 남은 기간(τ)라는 축을 추가할 수 있다. 이 경우 행사가격(forward moneyness)을 X축, 만기까지 남은 기간을 Y축, 그리고 내재변동성을 Z축에 놓으면 3차원 내재변동성 산포도가 그려지고 이 자료들을 바탕으로 3차원 곡면을 추정할 수 있다. 곡면 추정하는 방법에는 여러 가지가 있겠지만 이 보고서에서는 spline과 RBF를 사용하였다. 곡면을 추정하는 방법들에 관한 자세한 논의는 금융공학 VII: Scientific Computing for Finance and Economics[5]에서

Figure 11: Est. Put Option Price Curve



찾을 수 있다.

또한 내재변동성 곡면을 추정할 때에는 곡선 추정 때와는 다르게 행사가격을 forward moneyness로 환산하였다. forward moneyness의 개념에 대해서는 제1절에서 소개하였기 때문에 여기에서는 간략히 forward moneyness를 계산하기 위한 파이썬 코드를 Listing 4와 같이 제시한다. 코드에서 Forward_Moneyness 함수는 standardized log moneyness를 계산하기 위한 것이며, 현재가격을 미래가격으로 바꿔주는 역할은 Forward_Price 함수가 수행한다.

```

def Forward_Moneyness(target_value, t, S, K, T, r, Type, sigma):
    F = Forward_Price(t, S, T, r)
    tau = T - t
    return np.log(F/K) / (sigma*np.sqrt(tau))

def Forward_Price(t, S, T, r):
    return S*np.exp(r*(T-t))
```

Listing 4: Forward Moneyness

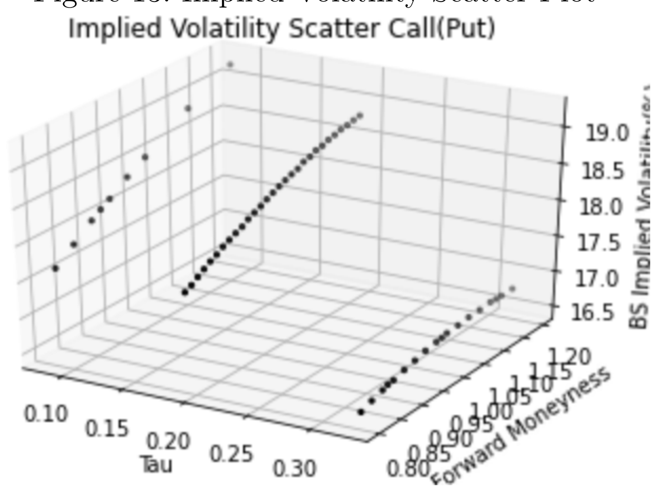
위 함수들을 통해 구한 forward moneyness를 FwdM 변수에 저장하면 surface를 그리기 위한 SPX 콜옵션 데이터는 Figure 12와 같이 정리된다.

Figure 12: Data for Implied Volatility Scatter Plot

	Midpoint	t	St	Strike	T	r	Type	Ivol	FwdM
0	443.25	0.386301	4155.86	3725.0	0.471233	0.039216	Call	0.106211	3.643699
1	420.45	0.386301	4155.86	3750.0	0.471233	0.039216	Call	0.187300	1.943658
2	396.20	0.386301	4155.86	3775.0	0.471233	0.039216	Call	0.185831	1.836329
3	373.35	0.386301	4155.86	3800.0	0.471233	0.039216	Call	0.195164	1.632464
4	350.10	0.386301	4155.86	3825.0	0.471233	0.039216	Call	0.196142	1.509603

이제 forward moneyness의 범위를 0.8에서 1.2로 제한한 다음 내재변동성을 만기까지 남은 기간(τ)과 forward moneyness에 대한 3차원 산포도로 나타내면 Figure 13과 같다.

Figure 13: Implied Volatility Scatter Plot



이제 다음과 같은 텐서곱(tensor product) B-Spline을 사용하여 내재변동성 곡면을 추정해보자. B-Spline에 대한 자세한 논의는 [5] 제4장을 참조할 수 있다.

$$s(x,y) = \Sigma_{j_1=0}^{q_1}\Sigma_{j_2=0}^{q_2}\theta_{j_1,j_2}B_{j_1,p_1}(x)B_{j_2,p_2}(y)$$

B-Spline을 사용하기 위해 scipy 패키지의 bisplrep 함수를 사용한다. 이를 위한 파이썬 코드는 Listing 5에 제시한다.

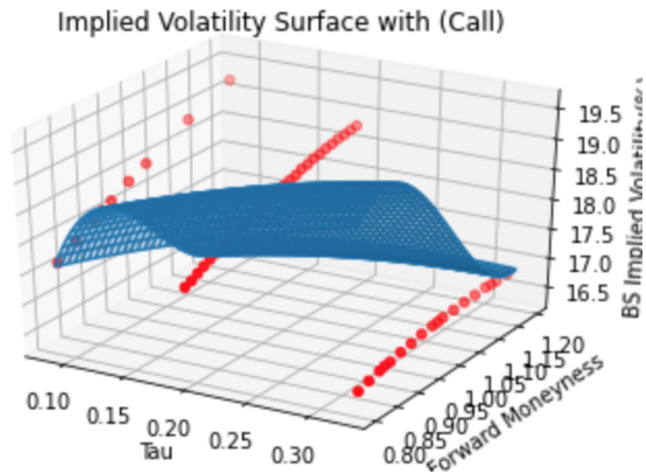
```

x = dfCall['tau'].copy()
y = dfCall['FwdM'].copy()
z = dfCall['Ivol'].copy()
xnew = np.linspace(x.min(), x.max(), len(x) * 3, endpoint=
    True)
ynew = np.linspace(y.min(), y.max(), len(y) * 3, endpoint=
    True)
xnew_edges, ynew_edges = np.meshgrid(xnew, ynew)

tck = interpolate.bisplrep(x, y, z, s=100)
znew = interpolate.bisplev(xnew, ynew, tck)

%matplotlib inline

fig = plt.figure()

ax = plt.axes(projection='3d')
ax.plot_wireframe(xnew_edges, ynew_edges, znew*100)
ax.plot_surface(xnew_edges, ynew_edges, znew*100, alpha=0.2)
ax.scatter3D(x,y,z*100, c='r')

ax.set_xlabel('Tau')
ax.set_ylabel('Forward Moneyness')
ax.set_zlabel('BS Implied Volatility(%)')
```

```python
plt.title("Implied Volatility Surface with (Call)")

plt.show()
```

Listing 5: B-Spline

그 결과 Figure 14과 같이 이해하기 힘든 곡면이 나타난다.

Figure 14: B-Spline Est.



이 보고서에서는 B-Spline을 직접 구현하는 대신 radial basis function interpolation(RBF)를 써서 내재변동성 곡면을 추정한다. RBF에 대해 위키피디아에서는 다음과 같이 설명한다.[16]

> Radial basis function (RBF) interpolation is an advanced method in approximation theory for constructing high-order accurate interpolants of unstructured data, possibly in high-dimensional spaces. The interpolant takes the form of a weighted sum of radial basis functions. RBF interpolation is a mesh-free method, meaning the nodes (points in the domain) need not lie on a structured grid,

and does not require the formation of a mesh. It is often spectrally accurate[1] and stable for large numbers of nodes even in high dimensions.

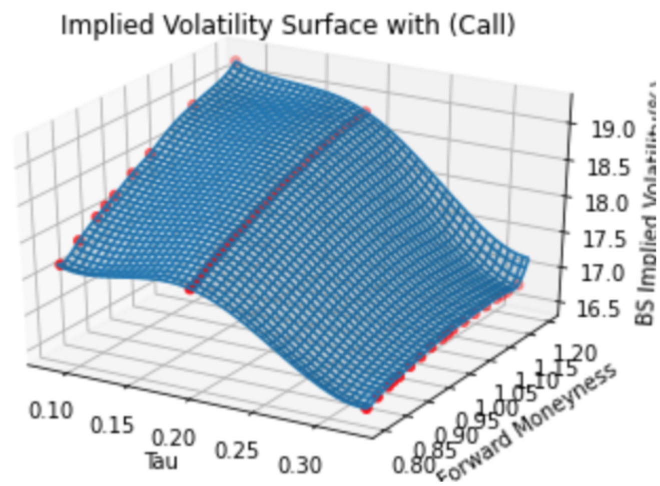
다음으로 RBF를 사용하기 Listing 6과 같이 코드를 작성하였다.

```
x = dfCall['tau'].copy()
y = dfCall['FwdM'].copy()
z = dfCall['Ivol'].copy()
rbf = interpolate.Rbf(x,y,z, function='thin_plate')

xnew = np.linspace(x.min(), x.max(), len(x) * 3, endpoint=
    True)
ynew = np.linspace(y.min(), y.max(), len(y) * 3, endpoint=
    True)
xnew_edges, ynew_edges = np.meshgrid(xnew, ynew)

znew = spline(xnew_edges, ynew_edges)
```

Listing 6: RBF

RBF를 통해 내삽한 결과는 Figure 15와 같다.

Figure 15: RBF Est.(call)



5 결론

S&P 500 Index 옵션 데이터를 블랙-숄즈식에 대입하여 내재변동성을 구해보았다. 이후 행사가격에 대해서 내재변동성 곡선을 추정하였으며, forward moneyness와 만기까지의 기간에 대해 내재변동성 곡면을 추정하였다. B-Spline을 통한 추정의 경우 결과가 좋지 못했지만 radial basis function interpolation을 사용한 결과 내재변동성 곡면 추정이 개선되었다.

References

[1] 최병선. *회귀분석 (上)*. (1997). 세경사.

[2] 최병선. (2004). *이산형 재무모형의 수리적 배경*. 세경사.

[3] 최병선. (2009). *금융공학I: Elements of Financial Engineering*. 세경사.

[4] 최병선. (2015). *금융공학 III: Introduction to Financial Engineering*. 세경사.

[5] 최병선. (2018). *금융공학 VII: Scientific Computing for Finance and Economics*. 세경사

[6] Choi, B., & Choi, M. Y. (2018). *General solution of the Black–Scholes boundary-value problem*. Physica A: Statistical Mechanics and its Applications, 509, 546-550.

[7] Choi, B., Kim, C., Kang, H., & Choi, M. Y. (2020). *General solutions of the heat equation.* Physica A: Statistical Mechanics and its Applications, 539, 122914.

[8] *Moneyness*. (2021, 6 7). Wikipedia.
https://en.wikipedia.org/wiki/Moneyness.

[9] Choi, B. (2021). *Volatility Models*. DATAly Lab Note 1.
https://drive.google.com/file/d/1rcXBKBgV7mStb49uCTNFcKhzOhw0oxqu/view.

[10] Choi, B. (2021). *A Study on Recursive Formulae*. DATAly Lab Note 5. https://drive.google.com/file/d/1u-ke3JbDWS-49TEFggmRgCHSYaE5sZLB/view.

[11] *What is forward moneyness and how to calculate it?* (2021, 6. 7). Quantatative Finance Stack Exchange.
https://quant.stackexchange.com/questions/43596/what-is-forward-moneyness-and-how-to-calculate-it.

[12] *Improving the way neural networks learn.* (2021, 6 7). Neural Networks And Deep Learning.

http://neuralnetworksanddeeplearning.com/chap3.html.

[13] *SPX 500 Option Prices.* (2021, 6 7). Barchart.

https://www.barchart.com/stocks/quotes/$SPX/options?expiration=2021-06-21-w&moneyness=allRows.

[14] *U.S. 10 Year Treasury Note.* (2021, 6 7). Market Watch.

https://www.marketwatch.com/investing/bond/tmubmusd10y?countrycode=bx.

[15] *Fed's New Inflation Targeting Policy Seeks to Maintain Well-Anchored Expectations.* (2021, 6 7). Federal Reserve Bank of Dallas. https://www.dallasfed.org/research/economics/2021/0406

[16] *Radial Basis Function Interpolation.* (2021, 6 7). Wikipedia.

https://en.wikipedia.org/wiki/Radial_basis_function_interpolation.