

(N/A)	15260.22	↑	4.04	↑	0.07%	0 (N/A)	3855	18	0.00%	0.00%
(N/A)	158.76	↑	2.58	↑	0.02%	0 (N/A)	2398	70	0.00%	0.00%
(N/A)	14758.04	↑	1.86	↑	1.63%	0 (N/A)	2446	98	0.00%	0.00%
(N/A)	7456.28	↑	0.10	↑	0.01%	0 (N/A)	-3530	121	0.00%	0.00%
(N/A)	13857.60	↑	1.42	↑	0.00%	0 (N/A)	2827	89	0.00%	0.00%
(N/A)	7058.67	↑	2.49	↑	0.01%	0 (N/A)	2659	80	0.00%	0.00%
(N/A)	26516.05	↑	2.05	↑	0.01%	0 (N/A)	3438	80	0.00%	0.00%
(N/A)	308705.76	↑	0.76	↑	0.00%	0 (N/A)	1459	105	0.00%	0.00%
(N/A)	17002.31	↑	2.31	↑	0.01%	0 (N/A)	274	79	0.00%	0.00%
(N/A)	13667.97	↑	1.97	↑	0.01%	0 (N/A)	1481	75	0.00%	0.00%
(N/A)	33148.24	↓	-1.76	↓	-0.01%	0 (N/A)	138	83	0.00%	0.00%
(N/A)	16666.59	↑	0.59	↑	0.00%	0 (N/A)	5804	4	0.00%	0.00%
(N/A)	514.28	↑	2.28	↑	0.44%	0 (N/A)	4999	114	0.00%	0.00%
(N/A)	669.78	↑	0.78	↑	0.12%	0 (N/A)	5873	54	0.00%	0.00%
(N/A)	584.55	↑	4.55	↑	0.78%	0 (N/A)	8	-8	0.00%	0.00%
(N/A)	820.70	↑	8.70	↑	1.06%	0 (N/A)	29	-29	0.00%	0.00%
(N/A)	132.40	↑	4.40	↑	3.32%	0 (N/A)	127	89	0.00%	0.00%
(N/A)	998.43	↓	-0.57	↓	-0.06%	0 (N/A)	86	-86	0.00%	0.00%
(N/A)	59.04	↑	5.94	↑	0.32%	0 (N/A)	61	-61	0.00%	0.00%

European Option Pricing by using Support Vector Regression

팀장 황상욱(경영 17)

팀원 길윤지(경금 19) 박수린(경금 19) 여현규(컴소 17)

이경원(건환공 16) 이성진(경금 17) 임지혜(경영 17)

목차

제 1장 서론 88

제 1절 연구배경

제 2절 선행연구

제 3절 연구목적

제 2장 본론 92

제 1절 이론적 배경

가. Black-Scholes 옵션 가격평가모형

나. 변동성 산출 한계

다. Machine Learning Model

제 2절 연구방법

가. SVR 변동성 추정 모델

나. SVR 가격 추정 모델

제 3절 연구결과

가. 내재변동성의 추정

나. 옵션가격의 추정

제 3장 결론 112

제 1절 연구한계

제 2절 선행연구의

참고 문헌 113

제 1장 서론

제 1절 연구 배경

오늘날 우리나라 증권시장에서는 하루 평균 60조원의 주식·채권·파생상품 등이 거래되고 있다. 경제규모가 커질수록 자본시장은 그 중요성이 더욱 커지기 때문에 현대 자본시장은 국가 경제의 필수 기반이자 그 뿌리라 볼 수 있다. 자본시장에는 크게 주식과 채권이 거래되는 현물시장과 선물, 옵션 및 스왑 등이 거래되는 파생상품시장으로 나뉜다. 과거 금융이 발전하면서 현물시장에서 거래되는 금융상품이 다양해졌고, 다양해진 금융상품은 더 큰 위험을 동반하게 되었다. 금융상품은 금리·주가·환율 등과 같은 외부 경제 상황에 시세가 수시로 변하기 때문에 금융이 발전하면서 이에 대비해야 하는 필요성도 커졌다. 이에 투자 전문가들은 통화나 증권 같은 상품에 투자를 하면서도 여러 변동성에 기인한 위험을 회피할 방도를 궁리하게 되었고, 그 결과 파생금융상품이 만들어졌다.

파생금융상품에는 선물, 옵션, 스왑, 주가연계증권(ELS), 주식워런트증권(ELW), 자산유동화증권(ABS) 등이 있으며 파생상품시장은 미래의 일정 시점 또는 일정 요건이 충족되면 행사할 수 있는 권리를 사고파는 시장이다. 옵션(option)은 이 중 '일정 시점'과 관련된 것으로, 만기일에 정해진 가격으로 어떠한 자산을 정해진 가격에 매입할 수 있는 권리인 콜 옵션(call option)과 매도할 수 있는 권리인 풋 옵션(put option)으로 나뉜다. 본 학술지는 유러피안 옵션에 대해 다루고 있는데, 이는 가장 간단한 옵션으로 옵션의 소지자에게 미리 정한 만기와 행사가격으로 한 단위의 기초자산을 사거나 팔 수 있는 권리를 주며 소지자가 약정 기간의 만기에만 권리행사를 할 수 있는 특징이 있다.

선물거래가 현물시장의 유동성에 미치는 영향에 관한 연구(옥기율, 2005)에 따르면 선물시장 거래량의 양적 증가는 현물시장의 유동성을 증가시키는 긍정적인 역할을 하여 결과적으로 자본시장 활성화에 기여한다. 자본시장은 이렇듯 시장을 구성하는 다양한 시장 사이의 긴밀한 관련 속에 움직이고 이렇게 만들어진 긴밀한 관련성은 자본시장의 효율성을 높이기 때문에 어느 금융당국에나 매우 중요한 과제로 여겨진다. 하지만 자본시장 활성화를 위해서는 먼저 적절한 금융상품 평가체계가 갖추어져야 한다.

옵션의 가격평가 방법에서 가장 널리 쓰이는 것은 1973년에 만들어진 블랙숄즈옵션가격평가 모형(Black&Scholes(1973))이다. 본 학술지에서는 기존 Black-Scholes 모형의 문제점에 착안해 더 나은 평가 방법의 필요성을 느껴 이에 대해 다루고자 한다.

또한 2021년 현재 빅데이터와 인공지능이 나날이 빠르게 발달하고 있다. The AI Index 2019 Annual Report(Stanford HAI, 2019)에 따르면 2010년대 들어 인공지능의 성능 향상 속도가 무어의 법칙보다 7배나 빠른 것으로 분석됐다. 이러한 흐름을 타고 많은 금융기관들이 AI를 금융자산 가격평가에 활용하고 있지만, 아직 이를 활용한 옵션가격평가 기술의 발전은 미미한 실정이다. 따라서 본 학술지는 머신러닝 기법을 활용하여 유러피안 옵션의 가격평가를 하는 것을 목표로 한다.

제 2절 선행연구

대표적인 옵션 가격 평가 모형으로는 단일 확률 분포(uniform distribution)를 가정한 Black-Scholes 모형(Black&Scholes(1973))과 Heston의 추계적 변동성 모형(S.Heston(1993))이 있다. 그중 Black-Scholes 공식을 사용해 유러피언 옵션의 이론가를 계산하기 위해서는 총 6개의 미지수가 필요하다. 변수(Variables)에는 기초자산의 가격(S)과 현재 시점(t) 뿐만 아니라 옵션의 행사가격(K), 만기시점(T), 무위험이자율(R), 그리고 기초자산의 변동성(σ) 등의 파라미터(parameters)가 있다. 백지현(2010)은 Black-Scholes 공식에서 필요한 기초자산의 수익률이 로그 정규분포에 따르며 그 변동이 등분산일 것이라고 하였다. 검증 결과에 따르면 기초자산의 수익률이 로그 정규분포를 따른다는 Black-Scholes 방정식의 가정은 타당한 것으로 보였지만, 등분산 변동 가정은 타당하지 않은 것으로 나타났다. 이에 대한 대안으로 AR(0)-IGARCH(1,1)모형이 기초자산의 변동을 나타내는 타당한 모형임을 나타냈다. 허현진(2010)은 금융 공학의 수치적 접근에 있어 중요하고도 기본적인 문제로 여겨지는 옵션 가격과 내재변동성에 대해 다뤘다. 유러피안 콜옵션에 Black-Scholes 모델을 적용한 식으로 출발하여 옵션 가격을 평가하는 유한차분법, 업윈드 방법, 그리고 유한 요소법을 구현하였다. 최종적으로 내재변동성을 예측하는 역시스템을 구상했다. 최현정(2020)은 딥러닝 기법을 활용한 KOSPI200 콜옵션의 가격을 예측한 연구를 진행했다. DNN으로 변동성을 제외한 나머지 다섯 가지 값을 학습하여 콜옵션의 가격을 추측했으며 LSTM 모형으로 내재변동성을 추정한 다음 공시된 정보를 사용하여 콜옵션의 가격을 계산했다. 또한, GARCH 모형으로 변동성을 추정한 다음 두 번째와 마찬가지로 콜옵션의 가격을 계산했다. 실험을 반복한 결과, 등가격 상태의 옵션 가를 계산함에 있어서 세 가지 방법 모두 높은 정확도를 보였다. 따라서 Deep Learning 기법을 사용하여 콜옵션의

가격을 추정한다면 필요 데이터가 Black-Scholes 모형에 비해 적다는 장점이 있으며 Black-Scholes 모형을 대체할 수 있을 것이라 분석했다.

기존 연구의 경우, 파생상품(유러피안 옵션)은 시간 및 기초자산의 가격 변화에 직접적으로 영향을 받는 상품이기 때문에 Black-Scholes 모형은 기본적으로 위의 두 가지 변수를 제외한 나머지 파라미터들은 고정된 상수라고 가정했다. 행사가격(K)과 만기 시점(T)은 어떤 옵션의 특정한 거래 조건으로 명시되어있기 때문에 위의 가정이 옳다. 하지만, 무위험이자율(r)과 변동성(σ)의 경우는 시장 상황이 변함에 따라 동태적으로 변화하는 값이기 때문에 변수로 가정하고 옵션의 가격을 산정해야 한다. 따라서, 기존 Black-Scholes 모형의 경우 잘못된 가정을 갖고 가격을 측정했기 때문에 정확도가 낮은 문제점이 나타난다.

기업 가격 수익률에 대한 분산은 사전적 자료(ex ante data)를 사용해야 하는 것이 옳지만 이를 정확하게 예측한다는 것은 사실상 불가능하다. 따라서, 기존 연구 논문에서는 이를 다소 안정적인 것으로 보아 사후적 자료(ex post data)를 이용하고 있다. Black-Scholes의 옵션 가격결정 모형의 문제점 중의 하나는 기초자산의 변동성이 만기일까지 일정하다는 가정인데 이러한 가정은 Kon(1984)과 Christie(1982) 등의 연구에 의해 현실과 거리가 있음이 밝혀졌다. 또한, 기존 연구의 경우 매월별 기업 가격을 월 말 자료를 토대로 산출했기 때문에 매월을 한 기로 하여 월간 기업 가격 수익률을 구하고 월간 기업 가격 수익률의 분산을 구한 다음 이를 연간으로 환산해 주는 방식을 취했다.

블랙숄즈(Black-Scholes) 방정식은 시간 역행적 특성을 가지고 있는 방정식 중 하나인 콜모고로프 후진 방정식(Kolmogorov Backward Equation)을 따른다. 즉, 시간 순서가 아닌 미래에서 현재로, 그리고 현재에서 과거로 후진하면서 발생하는 시스템 변화의 메커니즘을 설명하고 있다. 옵션의 가격과 기초자산의 변동성에 매우 민감하다는 점에서 옵션 모델의 적용 시 옵션 가격결정모형의 다른 변수들보다 변동성의 추정에 더 신경을 써야 할 것으로 보인다.

제 3절 연구 목적

현재 금융시장에는 주식, 채권 등의 기초자산 상품들과 이러한 기초자산을 이용한 파생상품 등의 다양한 상품들이 존재한다. 그중에서도 선물, 옵션 등은 수익 창출과 동시에 위험을 회피할 수 있는 금융상품으로 거래 시장이 활발하게 형성되어 있다. 옵션은 주식, 채권, 주가지수 등 특정 자산을 장래의 일정 시점에 미리 정한 가격으로 살 수 있는 권리와 팔 수 있는 권리를 매매하는 거래를 이야기한다. 옵션은 다양한 형태로 거래되고 있으며 이에 따라 옵션의 가격을 결정하는 방법도 다양하게 연구되고 있다.

본 학술지는 이 중 옵션가격의 대표적인 평가 방법인 Black-Scholes 모형을 기반으로 산출한 변동성 지수의 문제점에 주목했으며 변동성 오차를 최소화하기 위해 Support Vector Machine(SVM), Support Vector Regression(SVR)를 사용했다. Support Vector Machine(SVM)은 3차원을 넘어선 고차원 공간에서 원활하게 작동하고 비선형 분리 데이터를 커널 트릭을 사용하여 분류 모델링이 가능하다. Support Vector Machine(SVM)의 종류 중 하나인 Support Vector Regression(SVR)은 Vapnik(1995)에 의해 개발된 추정원리로서, 구조적 위험을 최소화함으로써 기존의 통계적 추정 방법이나 인공신경망보다 표본 외 예측 능력이 매우 뛰어난 것으로 알려져 있다. 최근 시계열 예측에 활발히 활용되기 시작하고 있으나 아직 적용 분야는 한정되어 있고, 특히 경제와 금융 분야에서의 응용연구는 적다. 본 학술지의 목적은, 기존 Black-Scholes 모형으로 산출할 수 없었던 내재 변동성을 Support Vector Regression(SVR)의 원리에 기반을 둔 새로운 예측 방법으로 유러피안 옵션의 가격 예측에 적용하는 것이다.

제 2장 본론

제 1절 이론적배경

가. Black-Scholes 옵션 가격평가모형

피셔 블랙(Fisher Black), 마이런 슐즈(Myron Scholes), 로버트 머튼(Robert Merton)에 의해 개발된 옵션 가격 결정 모형이다. 이 모형은 옵션의 가치를 평가하고 헷지하는 방식에 큰 영향을 미쳤으며 금융공학의 성장과 성공에 중요한 역할을 했다. Black-Scholes 방정식을 기초자산의 가격이 기하 브라운 운동¹에 따라 변화된다는 가정 하에 이용하면, 다음의 값들로 옵션 상품의 가격을 결정하는 모형을 도출할 수 있다.

S : 기초자산의 현재가격

K : 옵션 행사가격

T : 옵션 만기일

t : 옵션 현재시점

$z = (T - t)$: 옵션 만기일까지의 기간

r : 무위험 이자율

σ : 기초자산가격의 변동성

Black-Scholes는 계산의 용이성을 위한 기본 가정은 다음과 같다.

첫째, 잔존 기간 동안 가격의 변동성과 무위험 이자율은 일정하다.

둘째, 거래비용과 세금이 존재하지 않는 완전자본시장이다.

¹ 브라운 운동(Brownian Motion) : Appendix1 참고.

셋째, 옵션의 만기일까지 배당이 없다.

넷째, 증권의 거래는 연속적으로 일어난다.

또한, Black-Scholes 방정식은 기초자산이 이토폰정²

$$dS = \mu S dt + \sigma S dX \cdots \textcircled{1}$$

를 따른다는 가정하에 만들어졌다. 이 가정에 따라, 기초자산가격 S 와 시간 t 에 의존하는 무위험 포트폴리오 V 를 구성하고, V 의 가격인 $V(S, t)$ 의 미분 dV 는 이토폰정의 식에 의해

$$dV = \left(\frac{\partial V}{\partial S} \mu S + \frac{\partial V}{\partial t} + \frac{1}{2} \frac{\partial^2 V}{\partial S^2} \sigma^2 S^2 \right) dt + \frac{\partial V}{\partial S} \sigma S dX \cdots \textcircled{2}$$

를 따름을 알 수 있다. 여기서 가격 $V(S, t)$ 의 파생증권을 1단위를 사고 S 의 $\frac{\partial V}{\partial S}$ 단위를 파는 다음의 포트폴리오를 가정해보자. 이 포트폴리오의 가치 π 는

$$\pi = V(S, t) - \frac{\partial V}{\partial S} S$$

로 된다. 따라서 dt 시간동안 포트폴리오의 변화량은

$$d\pi = dV - \frac{\partial V}{\partial S} dS \quad \left(\frac{\partial V}{\partial S} \text{는 상수} \right)$$

이다. 이 식의 dV 와 dS 에 각각 $\textcircled{1}$, $\textcircled{2}$ 에 대입하여 정리하면

$$\begin{aligned} d\pi &= dV - \frac{\partial V}{\partial S} dS \\ &= \left(\frac{\partial V}{\partial S} \mu S + \frac{\partial V}{\partial t} + \frac{1}{2} \frac{\partial^2 V}{\partial S^2} \sigma^2 S^2 \right) dt - \frac{\partial V}{\partial S} \sigma S dX \\ &\quad - \frac{\partial V}{\partial S} (\mu S dt + \sigma S dX) \\ &= \left(\frac{\partial V}{\partial t} + \frac{1}{2} \frac{\partial^2 V}{\partial S^2} \sigma^2 S^2 \right) dt \cdots \textcircled{3} \end{aligned}$$

이다. 한편, 무위험 이자율(r)을 활용한 dt 시간 동안의 포트폴리오의 변화량은

² 이토 보조과정(Ito Lemma) : Appendix 참고.

$$d\pi = r \left(V(S, t) - \frac{\partial V}{\partial S} S \right) dt \quad \dots ④$$

로 성립된다. ③, ④로부터 Black-Scholes 미분방정식

$$\frac{\partial V(S, t)}{\partial t} = rV(S, t) - rS \frac{\partial V(S, t)}{\partial S} - \frac{\sigma^2 S^2}{2} \frac{\partial^2 V(S, t)}{\partial S^2} \quad \dots ⑤$$

를 도출할 수 있다. 이에 옵션의 만기시점에서 지불되는 지불금액함수를 경계조건으로 활용하면 다음의 식을 만족하게 된다.

$$V(S, z) = \begin{cases} S - E & \text{if } S \geq E \\ 0 & \text{if } S < E \end{cases} \quad \dots ⑥$$

나. 변동성 산출 한계

변동성의 정확도는 옵션의 이론 가격 측정에 큰 영향을 미치기 때문에 적절한 값을 계산해내는 것이 중요하다. 옵션의 가격을 산정할 때 Black-Scholes 모형에서 사용되는 모수들 중 기초자산의 가격(S), 행사가격(K), 현재시점(t), 만기시점(T), 무위험이자율(r)은 주어진 값인 반면, 옵션 기초자산의 수익률 변동성(σ)은 역사적 변동성(historical volatility)을 사용해왔다. 역사적 변동성은 과거 일정 기간 동안의 변수의 변화 값들을 이용해 계산한 추정치이며 통상적으로 수익률의 표준편차로 정의한다. 이를 반영하여 역사적 변동성을 나타낸 식은 다음과 같다.

$$HV = \sqrt{\frac{1}{n-1} \sum_{t=1}^n (R_t - \bar{R})^2}$$

$$\bar{R} = \frac{1}{n} \sum_{t=1}^n R_t$$

t 시점의 가격 = S_t , 일별 수익률 $R_t = \ln(S_t / S_{t-1})$

역사적 변동성을 산출할 때는 데이터 반영 기간에 주의해야 한다. 기간이 늘어나면 자료의 양이 많아진다는 장점이 있지만, 과거의 추이가 더 많이 반영되어 미래 예측의 적합성이 저해되므로 알맞은 설정이 중요하다. 또한, 역사적 변동성이 나타내는 과거 변동성은 미래 변동성과 동일시될 수 없다는 사실도 변동성의 정확도를 감소시킨다. 미래의 기초자산 가격이 과거의 기초자산 가격과 다른 것처럼 미래의 변동성과 과거의 변동성도 같을 수 없기 때문이다. 이처럼 역사적 변동성은 완전한 해결이 불가능한 문제점들을 가지고, HV를 활용하여 구한 옵션 이론가는 현재가와 상당한 괴리가 나타난다. 따라서 괴리를 보완하기 위해 고안된 개념인 내재 변동성에 대해 알아보겠다.

내재 변동성은 시장에서 거래되고 있는 옵션의 가격과 Black-Scholes 공식의 이론 가격이 동일하도록 만드는 변동성이라고 정의된다. 내재 변동성은 옵션의 시장가격으로부터 추출된 값이기 때문에 현재 시장에서 예상하고 있는 미래 기초자산 변동성의 컨센서스를 보여준다. 그러나 이러한 논리가 성립되기 위해서는 옵션의 시장가격이 균형가격이라는 점이 전제되어야 한다. 또한, 내재변동성에 관한 이해를 위해선 역사적 변동성과의 관계를 파악하는 것이 중요하다. 만약 역사적 변동성보다 내재 변동성이 높다면 옵션을 거래할 때 기대한 것보다 변동이 컸으므로 옵션가격이 고평가되어있음을 의미한다. 이를 토대로 내재 변동성이 현재부터 옵션 만기까지의 변동성을 측정할 수 있다는 점을 확인할 수 있다. 그런데 내재 변동성을 구하려면 옵션의 시장가격을 알아야 하고, 이를 Black-Scholes 옵션 모형에 역으로 통과시켜 산출해야 한다. 즉, 이미 만들어진 시스템의 출력값을 이용하여 역으로 입력값을 추정하는 과정을 거쳐야 하는데, 이를 리버스 엔지니어링(Reverse Engineering)이라고 부른다. 리버스 엔지니어링을 실현할 방법은 다양하지만, 본문에선 뉴턴-랩슨법에 대해 다뤄보겠다.

뉴턴-랩슨법은 역함수를 구할 수 없을 때 컴퓨터로 함숫값을 찾는 방법이다. 뉴턴의 방법은 $f(x) = 0$ 의 해를 구하기 위해, 근사해 x_n 에 대해 p 에 대한 방정식 $f(p+x_n) = 0$ 을 세우고 p 에 대해 차수가 높은 항을 제거하여 p 에 대한 일차식으로 만들어 푸는 것이다. 그런데 미분을 이용하여 $f(p+x_n)$ 을 일차 근사하면 $f(p+x_n) \approx f(x_n) + f'(x_n)p$ 이므로, $p = -\frac{f(x_n)}{f'(x_n)}$ 임을 알 수 있다. 다시 말해서, 방정식의 해를 구할 때 현재 x 값에서 접선을 그리고 접선이 x 축과 만나는 지점으로 x 를 이동시키면서 점진적으로 해를 찾는 방법이다. 뉴턴-랩슨법은 뉴턴법을 확장하여 연립방정식의 해는 물론 비선형 모델의 파라미터를 구하는 문제까지 확장할 수 있기 때문에 뉴턴-랩슨법을 통한 리버스 엔지니어링으로 내재 변동성을 구할 수 있다. 다만, 뉴턴-랩슨법 사용 시 함수가 미분 가능해야 하고, 시작점에서 값이 0이 되지 않는지 확인할 필요가 있다.

그러나 Black-Scholes 방정식을 이용하여 해를 역 추적하는 방법들은 모두 순환 논리에 빠진다는 문제점이 발생한다. 우선, 현재 시점부터 만기까지의 변동성을 알아야 옵션의 적절한 가격산정이 가능하다. 그러나 Black-Scholes 방정식을 변형하는 방법만을

이용한다면 만기까지의 변동성을 선제적으로 측정할 수 없고, 역산해서 변동성을 구해야 한다. 이러한 과정을 통해 산출된 변동성을 대입한 옵션 가격은 적정가격이 아니라 균형 상태의 시장에서 사후적으로 관찰한 값일 뿐이다. 실제로 동일한 기초자산에 대해 옵션 종류가 다양하게 존재하는데, 각 옵션으로부터 계산된 내재 변동성이 다르게 나타나는 경우가 많다. 옵션의 남은 기간과 행사가격에 따라 값이 달라지기 때문이다. 따라서 역추적으로 변동성을 구하는 과정은 옵션가격을 정확하게 설정할 수 없다고 판단했고, 괴리를 줄이기 위해 변동성을 사전적으로 측정할 수 있는 방법인 Support Vector Regression(SVR)을 활용하고자 한다. SVR은 Support Vector Machine(SVM)으로부터 파생된 회귀 식이므로 Machine Learning에 대한 이해가 선행되어야 한다.

다. Machine Learning Model

1959년 서 사무엘은 머신러닝(Machine Learning)을 “기계가 일일이 코드로 명시하지 않은 동작으로 데이터로부터 학습하여 실행할 수 있도록 하는 알고리즘을 개발하는 연구 분야”라고 정의했다. 이 정의에서 알 수 있듯, 머신 러닝과 일반적인 프로그램의 차이는 학습의 유무이다. 일반적인 프로그램의 경우 A는 B이다라는 조건을 인간이 제시하고 소프트웨어는 프로그램을 구동하는 역할을 한다. 반면 머신러닝(Machine Learning)의 경우 A는 B이다라는 조건을 인간이 제시하는 것이 아닌 소프트웨어가 스스로 데이터를 학습하고 패턴을 찾게 한다. 이러한 머신러닝(Machine Learning)의 학습은 지도 학습과 비지도 학습, 그리고 강화 학습으로 구성된다.

지도학습(Supervised Learning)의 경우 데이터 안의 인간이 그 데이터를 구분해 제시해준다. 그리고 이러한 구분된 데이터를 바탕으로 컴퓨터 스스로가 데이터를 구분해 내는 기준을 찾아내게 된다. 이번 학습에 사용될 SVM(Support vector machine)이나 결정 트리(Decision tree) 모델 등의 학습모델이 지도 학습을 이용하는 대표적인 예이다. 반면 비지도 학습(Unsupervised Learning)의 경우 데이터를 인간이 구분하지 않고 단지 데이터 그 자체만을 제시하여 컴퓨터가 처음부터 스스로 데이터를 구분해낸다. 비지도 학습의 예시로는 클러스터링(Clustering)이나 독립 성분 분석 (ICA; Independent Component Analysis) 등의 예시가 있다. 강화 학습(Reinforcement Learning)의 경우 보상을 통해 학습을 수행하는 머신러닝(Machine Learning) 방법이다. 강화 학습의 경우 상과 벌을 통해 학습을 수행한다. 좋은 행동에 대해서는 상을 주고, 나쁜 행동에 대해서는 벌을 준다. 이러한 학습을 통해 보상을 최대화하는 기준을 만들어 내는 것이 강화 학습의 목표이다. 이러한 강화 학습은 유전 알고리즘, 정보이론, 게임이론 등 다양한 분야에서 연구되고 있다.

이러한 머신 러닝(Machine Learning)은 여러 가지 형태로 활용되는데, 최근 화제가 되고 있는 딥러닝(Deep Learning) 또한 인공 신경망(artificial neural network)를 이용해 깊은 층(layer)을 만들어 내는 머신러닝 기법 중 하나이다. 이러한 인공 신경망(artificial neural network) 이외에도 여러 머신러닝(Machine Learning) 방법들이 있는데 이번 학술지에서는 여러 방법 중 지도학습 방법인 SVM(Support Vector Machine)이라는 방법을 다룰 것이다.

SVM(Support vector machine)³은 머신 러닝(Machine Learning)의 지도학습 종류 중 하나이다. SVM(Support vector machine)은 1995년 Vapnik와 Cortes에 의해 고안되었다. 이러한 SVM의 경우 다른 방법에 비해 수학적으로 분석하는 것이 수월하며, 조정해야 할 파라미터(Parameter)의 수가 많지 않아 비교적 간단하게 학습에 영향을 끼치는 요소들을 규명할 수 있었다. 또한 인공 신경망의 문제인 과적합 문제를 벗어날 수 있었기에 인공신경망(artificial neural network)을 제치고 90년대와 00년대 머신러닝(Machine Learning)의 주류로 자리 잡았었다. 그리고 이러한 특징을 바탕으로 이후 단순 분류를 넘어 SVM의 한 종류인 SVR(Support Vector Regression)이라는 방식을 사용한 회귀 문제의 영역까지 활용이 확장되었다. 이러한 SVM(Support vector machine)의 회귀와 분류를 활용한 방법들은 문서분류, 영상인식, 신용평가, 주가예측 등에서 뛰어난 결과를 보이며 다양한 분야에 성공적으로 적용되어 왔다.⁴ 하지만, SVM의 경우 자연어 처리 분야 등에서 한계를 가지기도 했다.⁵

SVM(Support vector machine)이 데이터를 분류해내는 방법은 기존 데이터를 바탕으로 해 데이터를 구분할 수 있는 구분선(Decision boundary line)을 만들어내는 것이다. 그리고 SVM(Support vector machine)을 통해 형성된 이러한 구분선은 기존 데이터뿐만 아니라 새로운 데이터를 구분해 낼 수 있는 기준이 된다. SVM(Support vector machine)의 경우 여러 가지 종류로 나뉘는데 구분선의 모양에 따라 Linear한 형태와 Nonlinear한 형태로 나눌 수 있다. 그리고 오차의 허용 유무에 따라 Soft margin과 Hard margin의 형태로 구분해 낼 수 있다.

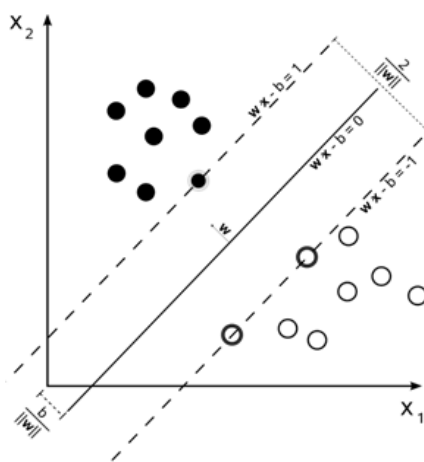
Linear SVM은 기본적인 SVM 모델이다. 이는 기본적으로 데이터가 선형으로 분리가 가능한 경우에 사용된다. 이는 커널을 이용하지 않고 기존 차원 내에서 데이터를 구분해 낼 수 있는 가장 간단한 형태의 분류 방법이다. 하지만 대부분 데이터의 경우 선형으로 구분될 수 없는 경우가 존재한다. Nonlinear하다는 것은 이러한 선형으로 구분될 수 없는 데이터를

³ 자세한 수학적 설명은 Appendix 2. 참고

⁴ 감정예측모형의 성과개선을 위한 Support Vector Regression 응용, 지능정보연구 제18권 제3호 2012년 9월(pp.185~202), 김성진, 유은정 정민규, 김재경, 안현철

⁵ 시각 인식 기술을 중심으로 한 인공지능 기술의 역사와 발전 방향, 최종현, 광주 과학 기술원, 방송과 미디어 제25권 1호 p14-18

커널(kernel)을 활용해 차원을 확장해 낸 이후 Nonlinear한 형태의 구분선을 만들어 내는 SVM(Support vector machine)을 말한다.



6

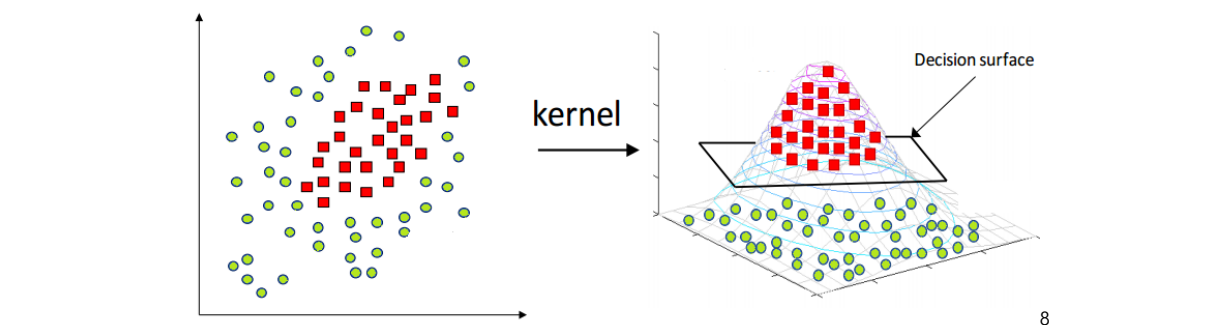
SVM(Support vector machine)의 경우 서포트 벡터(Support Vector)와 마진(Margin)의 개념을 이용해 최적의 구분선을 만들어 낸다. 서포트 벡터(Support Vector)란 서로 다른 데이터를 구분하는 경계에 있는 데이터들을 말한다. 그리고 이러한 서포트 벡터(Support Vector) 사이의 거리를 마진(Margin)이라고 한다. SVM의 경우 데이터를 분류할 때 최대의 마진(Margin)을 가지는 선형식을 구분선으로 설정하고 이 선형식을 기준으로 서로 다른 두 데이터를 분류해 낸다. 이러한 마진(Margin)을 바탕으로 서로 다른 데이터를 완전히 구분하는 SVM(Support vector machine)을 Hard Margin SVM이라고 부른다. 하지만 이러한 Hard Margin 방식으로 완전히 분류될 수 없는 경우가 존재한다는 것이 문제가 된다. 그리고 이러한 강한 기준을 바탕으로 경계를 정할 시 과적합(Overfitting) 문제가 발생할 수 있다는 점 또한 문제가 된다.

이러한 문제를 해결하기 위해서는 우선 1995년에 Corinna Cortes와 Vladimir N. Vapnik에 의해 제안된 Soft Margin SVM을 사용할 수 있다. Soft Margin SVM의 경우 Hard Margin 방식이 오차를 허용하지 않는 것과 달리 완벽히 데이터를 구분하기보다는 어느 정도의 오차를 허용하여 구분선을 설정한다. 이렇게 한다면 기존의 형태를 바탕으로 데이터를 구별해 낼 수 있다. 하지만, 이러한 Soft Margin 방법에서는 과소 적합(Underfitting) 문제가 발생할 수 있다는 점이 단점이다.

⁶https://miro.medium.com/max/838/1*gXvhD4lomaC9Jb37tzDUVg.png

그렇기에 이러한 Soft Margin 방법에서는 최대 마진(Margin) 뿐만 아니라 오차 정도를 최소화하는 목적 또한 적절히 고려해야 한다. 이를 위해 Soft Margin 방법에서는 오차를 나타내는 $\sum_{i=1}^n (\xi_i + \xi_i^*)$ 식에 C라는 변수를 추가하여 $C \sum_{i=1}^n (\xi_i + \xi_i^*)$ 로 적용함으로써 오차가 전체 식에 미치는 영향을 조정할 수 있게 한다. 이러한 C의 값을 추가함으로써 사용자는 C의 값을 적절하게 조정하여 오차의 영향을 적절한 비중으로 통제하여 과적합(Overfitting)의 정도를 조절할 수 있다. 즉, C의 값을 조정함으로써 사용자는 오차의 최소화와 최대 마진이라는 두 가지 목적 사이의 적절한 비중을 가지는 적절한 구별식을 산출할 수 있다는 것이다.

또한 이러한 방법 외에 Nonlinear한 선을 구분선으로 설정하여 데이터를 구분하는 방법 또한 있다. 이러한 Nonlinear한 구분선을 긋기 위해서는 기존 데이터의 형태를 변형하여 Linear한 형태로 구분 가능하게 해야 하며, 이를 위해 커널(kernel)⁷을 사용한다. 이러한 커널을 사용한다면, 기존 데이터가 가지고 있던 (x_1, x_2, \dots) 이라는 차원을 다양한 커널(kernel)을 통해 $(x_1, \sqrt{x_1 x_2}, \dots)$ 등과 같은 다양한 차원으로 변형할 수 있으며, 이러한 방법을 통해 기존에 Linear한 방법으로는 구분할 수 없었던 데이터들을 차원을 확장함으로써 확장된 차원 안에서 Linear한 형태로 구분할 수 있게 된다.

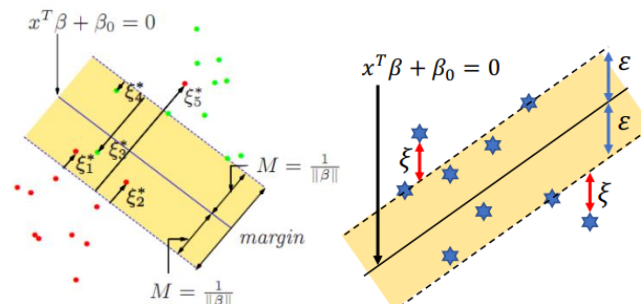


SVM은 Hard/Soft margin과 Linear/Nonlinear한 형태 이외에도 SVC(Support vector classification)와 SVR(Support vector regression)로 나뉜다. SVC와 SVR 모두 학습을 수행하는 방법은 동일하다. 하지만 두 가지 방법의 차이는 SVC와 SVR의 오차(ξ)의 정의에 의해 발생한다. SVC의 경우 오차(ξ)를 마진(Margin) 안에 포함된 점들 중 중심선(Decision boundary line)를 넘어선 데이터들만으로 Model Cost를 계산한다. 하지만, SVR의 경우 오차(ξ)는 마진(Margin)을 넘어선 데이터들의 마진(Margin)과의 거리로 산정한다. 이러한

⁷ 자세한 수학적 설명은 Appendix 3. 참고

⁸ https://miro.medium.com/max/838/1*gXvhD4lomaC9Jb37tzDUVg.png

특징 때문에 오차를 최소화하는 SVC의 경우 마진(Margin) 안에 있는 오차가 최소가 되도록 하고, SVR의 경우 마진(Margin) 밖에 있는 오차가 최소가 되도록 한다.



9

그림3) 좌측 SVC(Support vector classification) / 우측 SVR(Support vector regression)

이러한 이유 때문에 SVC(Support vector classification)의 경우 구분하기 위한 머신러닝(Machine Learning)에 가깝고 SVR(Support vector regression)의 경우 임의의 데이터를 회귀 함수를 통해 예측할 수 있는 방법에 가깝다. 그렇기에 SVC의 경우 일반적으로 종속변수가 범주형일 경우에 종류를 구별하기 위해 사용되는 경우가 많으며, SVR의 경우 종속변수가 연속형일 경우 회귀분석을 위해서 사용되는 경우가 많다.

이러한 SVM은 인공 신경망과 더불어 기계학습들 중에서도 예측력이 우수한 편이며, 과적합(Overfitting) 문제에 있어 어느 정도 자유롭기 때문에 텍스트 분류, 의학 등의 분야에서 많이 사용되고 있다. 하지만, 이러한 SVM 또한 인공 신경망(artificial neural network)에 비해 얕은 깊이를 가진 학습 방법이기에 전술하였듯 상대적으로 깊이 있는 학습을 필요로 하는 자연어 처리나 이미지 같은 시각에 관련된 문제에 있어 좋은 성과를 내지 못한다는 단점 또한 존재한다.

⁹ <https://todayisbetterthanyesterday.tistory.com/33>

제 2절 연구방법

정확한 European option의 가격을 예측하기 위하여 블랙-숄즈 모델을 기반으로, 고차원의 비선형 관계를 파악하기 위해 SVR model을 통해 기계학습으로 옵션 가격의 변동성 추정과 가격 추정을 진행했다. 추가적으로 RBF(Radial Base Function) kernel을 사용했다.¹⁰

블랙-숄즈 모델을 사용하였으므로, S, K, T, r, d, sigma, call option의 변수를 선언했다. 랜덤으로 1000개의 data를 생성하여 model training을 진행했다. 사용한 data의 요약 통계량은 다음과 같다.

Table1.

	S	K	T	r	d	sigma	price
count	1000	1000	1000	1000	1000	1000	1000
mean	49.02566	49.14345	1.55531	0.051985	0.048487	0.491227	10.726916
std	29.21374	33.796822	0.824199	0.028402	0.029098	0.286418	11.136775
min	0.463202	0.452814	0.1	0.000044	0.000019	0.000929	0
25%	23.597327	20.702109	0.85	0.028274	0.023191	0.244003	1.757082
50%	49.680738	45.247268	1.55	0.053977	0.047469	0.487395	7.457948
75%	74.431959	71.858639	2.27	0.076264	0.07388	0.741511	16.335121
max	99.971767	138.319466	2.99	0.099954	0.099762	0.997518	72.518066
skew	0.038786	0.570785	0.022643	-0.139569	0.073545	0.003092	1.436346
kurt	1.770899	2.541009	1.853263	1.872885	1.780557	1.777819	5.354623
j_b	63.196143	63.077385	54.877348	56.179433	62.861558	62.24023	574.858639

S는 기초 자산의 현재가격, K는 옵션 행사가격, T는 만기기간, r은 무위험 이자율, d는 배당률, sigma는 기초자산 가격 변화율의 표준편차를 의미하며 call option은 말그대로 call option일 경우 1, put option일 경우 0으로 설정했다.

```
F = S*np.exp((r-d)*T) # forward price
```

¹⁰ RBF kernel은 기본적으로 데이터 집합의 모든 점이 가설에 영향을 받는 방법이다. 선형으로 분리되지 않는 dataset인 경우, 비선형적인 무한의 차원으로 확장하여 data set을 분리하는 방식이다.


```

B = np.exp(-r*T) # discount factor

d1 = (np.log(F/K) + 0.5*sigma**2*T)/(sigma*T**.5)

d2 = (np.log(F/K) - 0.5*sigma**2*T)/(sigma*T**.5)

if call_option == 1:

    result = B*(F*N(d1) - K*N(d2))

if call_option == 0:

    result = B*(K*N(-d2) - F*N(-d1))

return B*result

```

위의 변수들을 바탕으로, 선물가격 S와 할인계수 B를 계산하고, call option 여부에 따라 가격을 산정 방식을 달리했다. 일련의 과정을 거쳐 생성된 options 변수와 가격의 예를 보이면 다음과 같다. 다만, 아래는 5개의 예시이며, 2단락에서 언급했듯, 1000개의 data를 랜덤으로 생성했다.

Table2.

	S	K	T	r	d	sigma	type	price
0	37.45401	35.95585	2.96	0.087205	0.016812	0.494521	0	5.354697
1	95.07143	57.99357	2.17	0.052723	0.047495	0.121695	0	0.007537
2	73.19939	81.25133	1.15	0.063254	0.005522	0.588094	1	15.78352
3	59.86585	77.22694	1.67	0.064447	0.094238	0.316078	0	19.19626
4	15.60186	21.37455	1.3	0.08198	0.007482	0.626968	0	6. 233007

가. SVR 변동성 추정 모델

1000개의 data를 가지고 본격적으로 SVR을 사용한 변동성 추정 모델을 만들었다. 변동성을 의미하는 sigma를 종속변수로 두고, 다른 parameters를 독립변수로 두었다.

SVR은 scale에 민감하기 때문에, 같이 StandardScaler() method를 통해 표준편차 scale로 통일하여 modeling 하였다.¹¹ 해당 과정의 코드는 아래와 같다.

```
#set y as the dependent variable, sigma

X,y = options[options.columns.difference(['sigma'])], options['sigma']

X_train, X_test, y_train, y_test = train_test_split(X, y, random_state=0)

y_train = y_train.values.reshape(-1,1)

y_test = y_test.values.reshape(-1,1)

scalerX = StandardScaler()

scalerX.fit(X_train)

X_train_scaled = scalerX.transform(X_train)

X_test_scaled = scalerX.transform(X_test)

scalerY = StandardScaler()

scalerY.fit(y_train)

y_train_scaled = scalerY.transform(y_train)

y_test_scaled = scalerY.transform(y_test)

reg = SVR()

reg.fit(X_train_scaled, y_train_scaled)
```

¹¹ 추후에, 역변환(inverse transform)과정을 통해, 실제 값으로 변환하는 과정을 거친다.

Grid Search이전, model의 결정계수는 train parameter가 R_square: 0.5973338536843651, test parameter가 R_square: 0.49837445467969166임을 파악했다. 결정계수 값이 유의미하다고 할 수 없다. Inverse_transform() method를 통해 실제 값으로 변환하는 과정을 거쳤다.

```
y_train_hat_scaled = reg.predict(X_train_scaled)

y_train_hat = scalerY.inverse_transform(y_train_hat_scaled)

print('R_square:', r2_score(y_train, y_train_hat))

y_test_hat_scaled = reg.predict(X_test_scaled)

y_test_hat = scalerY.inverse_transform(y_test_hat_scaled)

print('R_square:', r2_score(y_test, y_test_hat))

R_square: 0.5973338536843651

R_square: 0.49837445467969166
```

SVR 모델 사용을 위하여 모델의 오류 허용 범위를 나타내는 C, margin의 폭을 결정하는 epsilon, 결정 경계를 얼마나 유연하게 그을지를 결정하는 gamma, 그리고 각각, training과 testing에서 결정 계수인 r^2 (r square) 변수를 선언했다. 각각 변수는 아래와 같은 선택지를 갖는다. 선택지의 산출 기준은 SVR의 보편적인 계수들로 선정했다.

```
C_ep_ga_data = pd.DataFrame(columns = ('C', 'epsilon', 'gamma', 'training r2', 'test r2'))

C_settings = [1, 100]

epsilon_settings = [0.001, 0.01, 0.1]
```

```
gamma_settings = [0.01, 0.1]
```

위 12가지 경우를 Grid Search를 통해 결정 계수를 구하면 아래와 같다.

Table3.

	C	epsilon	gamma	training r2	test r2
0	1	0.001	0.01	0.20431	0.194981
1	1	0.001	0.1	0.540725	0.475233
2	1	0.01	0.01	0.204612	0.195448
3	1	0.01	0.1	0.540839	0.476346
4	1	0.1	0.01	0.205281	0.198945
5	1	0.1	0.1	0.543887	0.466384
6	100	0.001	0.01	0.597981	0.567637
7	100	0.001	0.1	0.859926	0.693819
8	100	0.01	0.01	0.598332	0.569101
9	100	0.01	0.1	0.860687	0.697323
10	100	0.1	0.01	0.597275	0.570019
11	100	0.1	0.1	0.865319	0.707075

12가지 경우 중, 결정 계수가 가장 큰 index11의 계수를 SVR model에 사용했다.
 추가적으로, 계수의 RMSE(Root Mean Square Error)값을 구해 모델의 적합성을 평가했다.

Table4.

	C	epsilon	gamma	training rmse	test rmse
0	1	0.001	0.01	0.066317	0.062497
1	1	0.001	0.1	0.038278	0.04074
2	1	0.01	0.01	0.066291	0.062461

3	1	0.01	0.1	0.038269	0.040654
4	1	0.1	0.01	0.066236	0.062189
5	1	0.1	0.1	0.038015	0.041427
6	100	0.001	0.01	0.033506	0.033566
7	100	0.001	0.1	0.011674	0.02377
8	100	0.01	0.01	0.033477	0.033453
9	100	0.01	0.1	0.011611	0.023498
10	100	0.1	0.01	0.033565	0.033381
11	100	0.1	0.1	0.011225	0.022741

역시, index11이 편차가 제일 작기 때문에 적합하다 할 수 있다. 따라서, $C = 100$, $\epsilon = 0.100$, $\gamma = 0.10$ 의 SVR model을 선택했다. 위의 모델을 통해 구한 sigma값과 실제 sigma값 비교를 통해 모델의 성능을 파악했다. 결과는 연구결과 항목에서 다루도록 하겠다.

나. SVR 가격 추정 모델

변동성 추정 모델과 비슷한 과정을 거친다면, 가격 추정 모델의 계수를 결정할 수 있다. 다만, 가격추정 모델이므로 price를 종속변수로 설정하고 이외의 변수를 독립변수로 설정한다.

```
#set y as the dependent variable, price
```

```
X,y = options[options.columns.difference(['price'])], options['price']
```

Grid Search 이전의 결정 계수는 다음과 같다.

```

y_train_hat_scaled = reg.predict(X_train_scaled)

y_train_hat = scalerY.inverse_transform(y_train_hat_scaled)

print('R_square:', r2_score(y_train, y_train_hat))

y_test_hat_scaled = reg.predict(X_test_scaled)

y_test_hat = scalerY.inverse_transform(y_test_hat_scaled)

print('R_square:', r2_score(y_test, y_test_hat))

R_square: 0.9480723411715527

R_square: 0.9244297418392151

```

SVR 모델의 계수들을 역시 Grid Search를 통해 가장 적합한 모델을 찾는다. 모델의 계수 후보군은 역시 변동성 추측모델과 동일하게 아래와 같이 설정한다.

```

C_settings = [1, 100]

epsilon_settings = [0.001, 0.01, 0.1]

gamma_settings = [0.01, 0.1]

```

Grid Search의 결과는 아래와 같다. Index7의 결과가 가장 큰 결정 계수 값을 가짐을 알 수 있다. 따라서, $C = 100$, $\epsilon = 0.001$, $\gamma = 0.10$ 임을 알 수 있다. 해당 모델은 결정 계수가 0.999620으로 상당히 큰 연관성을 나타낸다.

Table5.

	C	epsilon	gamma	training r2	test r2
0	1	0.001	0.01	0.504558	0.508406
1	1	0.001	0.1	0.942616	0.930367
2	1	0.01	0.01	0.506435	0.510814
3	1	0.01	0.1	0.942975	0.930497
4	1	0.1	0.01	0.511139	0.516726
5	1	0.1	0.1	0.939038	0.924168
6	100	0.001	0.01	0.97806	0.970134
7	100	0.001	0.1	0.99962	0.991558
8	100	0.01	0.01	0.977992	0.969671
9	100	0.01	0.1	0.99961	0.990807
10	100	0.1	0.01	0.975588	0.968554
11	100	0.1	0.1	0.994643	0.979673

RMSE의 결과 값 역시 아래와 같이 index7의 결과가 가장 큼을 확인할 수 있다.

Table6.

	C	epsilon	gamma	training rmse	test rmse
0	1	0.001	0.01	61.406951	60.594834
1	1	0.001	0.1	7.11237	8.58315
2	1	0.01	0.01	61.174257	60.297921
3	1	0.01	0.1	7.06783	8.567099
4	1	0.1	0.01	60.591177	59.569258
5	1	0.1	0.1	7.555865	9.347216
6	100	0.001	0.01	2.719328	3.681338
7	100	0.001	0.1	0.047147	1.040569
8	100	0.01	0.01	2.727804	3.738435
9	100	0.01	0.1	0.048301	1.133143
10	100	0.1	0.01	3.025714	3.876124
11	100	0.1	0.1	0.663963	2.505494

해당 모델을 통해, SVR 가격 추측 모델을 통해 예측한 price와 실제 price의 비교는 연구결과 항목에서 다루도록 하겠다.

제 3절 연구결과

가. 내재변동성의 추정

Support Vector Regression을 활용하여 옵션의 내재 변동성과 옵션의 가격을 추정해보았다. 내재 변동성의 추정에 앞서 모델을 효율적으로 적합시키기 위하여 Grid Search를 진행하였다. Grid Search를 통해 Support Vector Regression의 파라미터인 C, Epsilon, Gamma를 조절하여 결정 계수를 최대화하고 RMSE(평균 제곱 오차)를 최소화하는 지점을 찾을 수 있다. 여기서 결정 계수는 수정 결정 계수를 사용하지 않고 일반 결정 계수를 사용한다.

$$R^2 = 1 - \frac{SSR}{SST}$$

$$RMSE(\hat{\theta}) = \sqrt{MSE(\hat{\theta})} = \sqrt{E((\hat{\theta} - \theta)^2)}$$

내재 변동성 추정에서의 Grid Search 이전의 결정 계수는 0.4984였으나, Grid Search를 통해 찾은 극대구간에서의 결정 계수는 0.8653으로 약 67.8% 정도의 적합도 향상이 이루어졌다. 결정 계수가 0.8653인 구간에서의 C, epsilon, gamma는 각각 100.0, 0.1, 0.1로 도출이 되었다. 마찬가지로 RMSE가 0.1122로 최소인 구간에서의 C, epsilon, gamma 역시 100.0, 0.1, 0.1로 나타났다. 데이터를 8대 2로 분할하여 80%를 학습 데이터로 사용하고 20%를 테스트 데이터로 사용하였다. 분할한 데이터를 Support Vector Regression을 통해 학습함에 있어 커널 함수는 RBF 함수를 사용하였고, C, epsilon, gamma는 위에서 도출한 결정 계수를

최대화하고 RMSE를 최소화하는 지점을 활용하였다. Support Vector Regression을 통해 추정된 내재 변동성(pred_sigma)는 아래와 같다.

Table.7

S	K	T	r	d	sigma	pred_sigma	type	price
37.4540	35.955851	2.96	0.087205	0.016812	0.494521	0.3941	0	5.354697
95.0714	57.993573	2.17	0.052723	0.047495	0.121695	0.2214	0	0.007537
73.1994	81.251328	1.15	0.063254	0.005522	0.588094	0.5426	1	15.783524
59.8658	77.226944	1.67	0.064447	0.094238	0.316078	0.3801	0	19.196255
15.6019	21.374554	1.3	0.08198	0.007482	0.626968	0.6171	0	6.233007

위의 결과를 통한 해석은 3가지로 가능하다. 옵션의 변동성은 대체로 등가격, 내가격, 외가격 상태에 따라 나누어진다. 등가격이란 현재 주식가격과 옵션의 행사가격이 비슷한 수준에 위치해 있는 경우를 의미한다. 만약 주식의 가격이 10,000원인데, 옵션의 행사가격이 9,900원이라면 그 옵션은 등가격 상태에 있다고 표현한다. 내가격은 현재 옵션의 내재가치가 존재하는 상태를 일컫는다. 즉, 옵션을 행사할 경우 이익이 발생한다면 그것은 내가격 상태에 있는 옵션이다. 콜옵션의 경우 기초자산의 가격보다 행사가격이 더 낮은 경우가 내가격 상태이다. 반대로 풋옵션은 기초자산의 가격이 행사가격보다 낮은 경우가 내가격 상태이다. 외가격은 현재 옵션에 내재가치가 없는 상태를 말한다. 즉, 옵션을 당장 행사할 경우 이익이 발생하지 않게 된다. 내재 변동성은 옵션이 등가격에 있을 때 가장 크고, 외가격이나 내가격으로 갈수록 작아지게 된다. 위의 표에서도 이러한 현상이 나타나고 있다. 그뿐만 아니라 옵션의 잔존만기가 긴 경우에는 역사적 변동성과 실제 변동성 간의 괴리가 심하게 나타나고 있다. 즉, 만기가 길게 남을수록 변동성에 베팅을 한 차익거래 포착이 가능하다. 옵션의 변동성의 실제와 예상의 괴리가 클수록 시장가격에 왜곡이 크게 발생하기 때문이다. 적합에서의 편향이 발생하지 않았다는 가정하에 Support Vector Regression을 사용하여 추정된 내재 변동성은 예측의 정확도를 보장받을 수 있다. 적합 편향을 측정하기 위해 EVS(Explained Variance Score; 설명분산점수)를 통해 확인을 해보았다.

$$EVS = 1 - \frac{SSR - \text{Mean Error}}{SST}$$

결정 계수와 유일의 차이점은 분자에서 Mean Error를 제거한 부분이다. 따라서 결정 계수와 EVS값이 크게 차이가 난다면 오차에 편향이 존재한다는 것을 의미하고, 이는 적합이 제대로 이루어지지 않았음을 의미하게 된다. 결정 계수는 0.8653이고 EVS는 0.8985로, 약 0.03 정도의 차이는 편향이 거의 없음을 의미하므로 적합이 잘 이루어졌음을 나타낸다.

나. 옵션가격의 추정

내재 변동성의 추정과 비슷한 방법으로 추정을 하였다. Price를 제외한 부분을 독립변수로 두고 Price 부분을 종속변수로 하여 학습을 진행하였다. Grid Search 역시 내재 변동성을 추정하는 방식과 동일하게 진행을 하여 결정 계수를 최대화하고 RMSE를 최소화하는 지점을 도출하였다. 그 결과, C, epsilon, gamma가 각각 100.0, 0.01, 0.1 일 때, 결정 계수는 0.9996이고, RMSE가 0.0483으로 나타났다. Support Vector Regression을 사용하여 추정한 옵션가격(pred_price)는 다음과 같다.

Table8.

S	K	T	r	d	sigma	type	price	pred_price
37.4540	35.955851	2.96	0.087205	0.016812	0.494521	0	5.354697	<u>5.3558471</u>
95.0714	57.993573	2.17	0.052723	0.047495	0.121695	0	0.007537	<u>0.0084168</u>
73.1993	81.251328	1.15	0.063254	0.005522	0.588094	1	15.783524	<u>15.784696</u>
59.8658	77.226944	1.67	0.064447	0.094238	0.316078	0	19.196255	<u>19.197503</u>
15.6019	21.374554	1.3	0.08198	0.007482	0.626968	0	6.233007	<u>6.2341997</u>

결정 계수가 0.9996이었기 때문에 실제 가격과 추정가격 간의 괴리가 크지 않다. 등가적이고 변동성이 클수록 옵션의 가격이 높게 산정되는 경향을 띄고 있으므로 실제 옵션가격의 성질과 일치한다. 따라서 옵션가격 추정을 할 때, 역사적 변동성이 아닌 Support Vector Regression을 통해 예측한 내재 변동성을 산입하여 계산한다면 기존 옵션 가격평가의 한계를 극복함으로써 만기시점까지의 변동성을 고려한 적정 옵션가격을 산출할 수 있다. 동시에 옵션 가격의 변동을 파악하기 위한 Delta, Theta, Gamma, Vega, Rho와 같은 파라미터들 역시 당위성을 부여받을 수 있다.

제 3장 결론

제 1절 연구한계

첫째, 기초자산의 가격과 이자율의 변동을 만기시점까지 시뮬레이션을 통해 예상하여야 한다. 기초자산의 가격과 이자율의 변동에 대한 예측이 담보되지 않는다면 SVR의 변수로 과거 기초자산의 가격과 이자율을 사용하기 때문에 현재 옵션 가치에 대한 설명만 가능할 뿐 미래 특정 시점에 대한 옵션가격 예측에는 어려움이 있다. 둘째, 블랙숄즈 방정식은 옵션이 유럽형 옵션이라는 것을 전제하고 있기 때문에 미국형 옵션에 대해서는 옵션의 내재 변동성을 산출하기 까다로워진다는 한계가 있다. 만기에만 권리를 행사할 수 있는 유럽형 옵션과 달리, 미국형 옵션은 만기 이전이더라도 권리를 행사할 수 있기 때문에 이항모형을 사용하여 가치를 평가하게 된다. 변수들을 연속적으로 시뮬레이션 할 수 없기 때문에 방정식이 존재하지 않고, 다양한 추정기법을 통해 옵션의 가격을 추정하게 된다. 따라서 미국형 옵션에서는 Support Vector Regression을 활용하여 옵션의 내재 변동성과 가격을 합리적으로 추정할 수 없다는 문제가 있다.

제 2절 연구의의

옵션가격을 평가할 때 블랙숄즈 방정식이나 이항모형을 사용할 때 늘 한계로 지적되었던 내재 변동성의 산출을 합리적으로 하게 되었다는 것에 그 의의가 있다. 본 연구에서는 옵션의 내재 변동성 추정과 옵션의 가격 추정을 독립적으로 시행하였다. 그 결과, 내재 변동성의 추정에서는 역사적 변동성과 Support Vector Regression을 통해 추정한 변동성의 괴리가 상당했다. 또한, 옵션가격평가의 경우에 블랙숄즈 방정식을 통해 계산한 가격과 Support Vector Regression을 활용하여 계산한 가격의 차이가 없었다. 따라서, 내재 변동성 추정을 통해 얻은 내재 변동성을 블랙숄즈 방정식의 변수로 산입하여 가격을 평가한다면 역사적 변동성을 사용할 때 보다 합리적으로 옵션의 가격을 산정할 수 있다. 데이터의 분류를 목적으로 쓰이던 Support Vector Machine의 아이디어를 회귀분석에 활용함으로써 복잡한 차원의 표본들에 대해 설명력이 있는 회귀선을 도출할 수 있었다. 이를 통해 기초자산의 만기까지의 변동성을 예측할 수 있는 토대를 마련하였다.

VI. 참고문헌

옥기울. (2005). 선물거래가 현물시장의 유동성에 미치는 영향에 관한 연구. *선물연구*, 13(1), 25-43. The AI Index 2019 Annual Report(Stanford HAI, 2019)

임헌세. (2020). 기계학습 기반 비모수적 옵션 가격 평가 모형의 코스피 200 지수 옵션에 대한 적용= Application of non-parametric option pricing model based on machine learning for KOSPI200 index option.

황창하, 최상일, & 심주용. (2018). 심층 다중 커널 최소제곱 서포트 벡터 회귀 기계. *한국데이터정보과학회지*, 29(4), 895-902.

한규식. (2020). 마코프 체인 근사법을 이용한 일반화 블랙-숄즈 모형 하의 옵션 가격 측정. *한국경영과학회지*, 45(3), 11-25.

Song, H. Derivation of Black-Scholes equation 블랙-숄즈 방정식 유도과정.

Black-Scholes 방정식의 수치해석 입문 /정다래, 김준석 공저.

Awad, M., & Khanna, R. (2015). Support vector regression. In *Efficient learning machines* (pp. 67-80). Apress, Berkeley, CA.

Smola, A. J., & Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and computing*, 14(3), 199-222.

Andreou, P. C., Charalambous, C., & Martzoukos, S. H. (2009, September). European option pricing by using the support vector regression approach. In *International Conference on Artificial Neural Networks* (pp. 874-883). Springer, Berlin, Heidelberg.

MacBeth, J. D., & Merville, L. J. (1979). An empirical examination of the Black-Scholes call option pricing model. *The journal of finance*, 34(5), 1173-1186.

Noble, W. S. (2006). What is a support vector machine?. *Nature biotechnology*, 24(12), 1565-1567.

VII. 부록

브라운 운동과 이토의 보조정리

① 브라운 운동

금융 문제와 같이 시간과 연동된 변동에 대한 수학적 모델은 대개 브라운 운동과 같은 확률과정을 사용하여 나타낸다. 브라운 운동의 엄밀한 수학적 기초를 닦은 미국의 수학자 Nobert Wiener의 이름을 따서 브라운 운동을 위너과정이라고도 한다. 금융시장에서 기초자산의 가격경로는 위너과정을 따른다고 가정하는 경우가 많다.

Brown 운동

다음 조건들을 만족하는 확률과정 $\{S(t)|t \leq 0\}$ 을 Brown 운동이라고 부른다.

- $S(0) = 0$
- 임의의 $0 < t_1 < t_2 < t_3 \dots$ 에 대해서 $S(t_1), S(t_2) - S(t_1), S(t_3) - S(t_2), \dots$ 는 서로 독립이다.
- 만일 $0 \leq \alpha \leq \beta$ 이면, $S(\beta) - S(\alpha)$ 는 평균이 0이고 분산이 $\sigma^2(\beta - \alpha)$ 인 정규분포를 따른다. 이 σ 를 변동성(volatility)라 한다.
- 확률과정 $\{S(t)\}$ 에서 실현된 표본경로는 t 의 연속함수이다.

즉, σ 가 1일 때의 확률과정을 표준 Brown 운동이라 한다.

일반화 Brown 운동

만일 확률변수 $S(t)$ 가 위의 Brown 운동의 첫번째, 두번째, 그리고 세번째 조건을 만족하고 평균이 μt 이고 분산이 $\sigma^2 t$ 인 정규분포를 따르면 일반화 Brown 운동이라 부른다. 이 μ 를 추세모수(drift parameter)라 한다.

일반화 Brown 운동에서 파생된 기하 Brown 운동은 다음과 같이 정의된다.

기하 Brown 운동(Geometric Brownian Motion, GBM)

일반화된 Brown 운동 $\{S(t)|t \leq 0\}$ 에 대해서, 식 $Z(t) = e^{S(t)}$ 로 정의된 $\{Z(t)|t \leq 0\}$ 를 기하 Brown 운동이라 한다.

② 이토의 보조정리(Ito Lemma)

시계열 $S(t)$ 의 변화량 dS 가 다음의 식에 따라 움직이고 있다고 하자. dX 는 정규분포에서 나온 표본이다.

$$dS = a(S, t)dt + b(S, t)dX$$

일반화된 위너 과정의 상수 a 와 b 를 S 와 t 에 대한 함수 $a(S, t), b(S, t)$ 로 일반화한 것을 이토과정이라고 한다. 따라서, 이 시계열 $S(t)$ 의 움직임을 이토 과정이라고 할 수 있다. 이 과정을 이용하면 이토의 보조정리(Ito Lemma)를 이끌어낼 수 있다.

S 가 이토과정

$$dS = a(S, t)dt + b(S, t)dX$$

를 따를 때, S 와 t 의 함수 $V(S, t)$ 의 동향은

$$dV = \left(\frac{\partial V}{\partial t} + \frac{1}{2} \frac{\partial^2 V}{\partial S^2} b^2(S, t) + \frac{\partial V}{\partial S} a(S, t) \right) dt + \frac{\partial V}{\partial S} b(S, t) dX$$

를 따른다.

(증명) 함수 V 의 변화량 dV 는 2변수 함수의 테일러 전개를 이용하면

$$\begin{aligned} dV &= V(S + dS, t + dt) - V(S, t) \\ &= \frac{\partial V}{\partial S} dS + \frac{\partial V}{\partial t} dt + \frac{1}{2} \frac{\partial^2 V}{\partial S^2} (dS)^2 + \frac{\partial^2 V}{\partial S \partial t} dS dt + \frac{1}{2} \frac{\partial^2 V}{\partial t^2} (dt)^2 + \dots \end{aligned}$$

로 나타난다.

S 가 이토과정을 따르므로, dS 에 ' $dS = a(S, t)dt + b(S, t)dX$ '를 대입하면

$$\begin{aligned}
 dV &= \frac{\partial V}{\partial S}(a(S, t)dt + b(S, t)dX) + \frac{\partial V}{\partial t}dt \\
 &\quad + \frac{1}{2} \frac{\partial^2 V}{\partial S^2} (a(S, t)dt + b(S, t)dX)^2 \\
 &\quad + \frac{\partial^2 V}{\partial S \partial t} (a(S, t)dt + b(S, t)dX)dt + \frac{1}{2} \frac{\partial^2 V}{\partial t^2} (dt)^2 + \dots \\
 &= \frac{\partial V}{\partial S} a(S, t)dt + \frac{\partial V}{\partial S} b(S, t)dX + \frac{\partial V}{\partial t} dt + \frac{1}{2} \frac{\partial^2 V}{\partial S^2} a^2(S, t)(dt)^2 \\
 &\quad + \frac{1}{2} \frac{\partial^2 V}{\partial S^2} b^2(S, t)(dX)^2 + \frac{\partial^2 V}{\partial S^2} a(S, t)b(S, t)dtdX + \frac{\partial^2 V}{\partial S \partial t} a(S, t)(dt)^2 \\
 &\quad + \frac{\partial^2 V}{\partial S \partial t} b(S, t)dtdX + \frac{1}{2} \frac{\partial^2 V}{\partial t^2} (dt)^2 + \dots
 \end{aligned}$$

가 된다. $dt \rightarrow 0$ 일 때의 관계식 $dX = \sqrt{dt}$ 를 사용하면 $(dt)^2$ 과 $dtdX$ 와 같이 1차보다 큰 항은 급속히 0에 가까워지므로 이들 항이 포함된 식들을 무시할 수 있다. $(dX)^2$ 의 항은 dt 가 된다. 따라서 위 식은 다음과 같이 이토 과정으로 나타낼 수 있다.

$$dV = \left(\frac{\partial V}{\partial S} a(S, t) + \frac{\partial V}{\partial t} + \frac{1}{2} \frac{\partial^2 V}{\partial S^2} b^2(S, t) \right) dt + \frac{\partial V}{\partial S} b(S, t)dX$$

SVM의 수학적 표현¹²

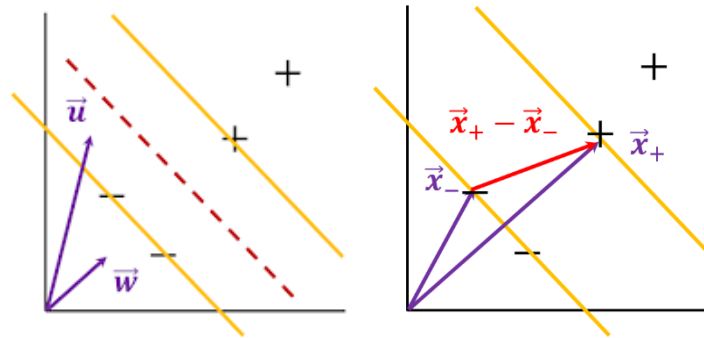


그림 4) SVM(support vector model)

$\vec{w} \cdot \vec{u} + b \geq 0$ 을 위 빨간 선이라고 가정할 때, $\vec{w} \cdot \vec{u} + b \geq 0$ then '+' (1)

이를 조금 구체화해서 표현할 때, 노란 선을 $\vec{w} \cdot \vec{x} + b = \pm 1$ 로 가정

$y_i = \vec{w} \cdot \vec{x} + b$ 일 시, $y_i = 1$ for '+', $y_i = -1$ for '-' (2)

위 (1)식과 (2)식을 고려할 시, $y_i(\vec{w} \cdot \vec{x} + b) \geq 1$

이는 $y_i(\vec{w} \cdot \vec{x} + b) - 1 \geq 0$ (3)로 표현할 수 있음.

(3)을 고려 시 위쪽의 노란 선을 $y_i(\vec{w} \cdot \vec{x} + b) - 1 = 0$ (4.1)로 표현할 수 있음

동일한 방법을 반복한다면 아래쪽 노란 선의 경우 $y_i(\vec{w} \cdot \vec{x} + b) + 1 = 0$ (4.2)

SVM의 경우 Margin을 최대화하는 구분선을 찾아내는 방법임. 즉 선 사이의 넓이를 구해야 함

선 사이의 넓이는 $WIDTH = (\vec{x}^+ - \vec{x}^-) \cdot \frac{\vec{w}}{\|\vec{w}\|}$ 로 표현할 수 있음

(4.1)과 (4.2)식을 통해 $(\vec{x}^+ - \vec{x}^-)$ 를 정리한다면 $WIDTH = \frac{2}{\|\vec{w}\|}$ - (5)로 정리됨

¹² <https://leejiyoon52.github.io/Support-Vector-Regression/>, <http://jaejunyoo.blogspot.com/2018/01/support-vector-machine-1.html>

WIDTH를 최대화하는 문제는 $\frac{1}{\|\vec{w}\|}$ 로 정리될 수 있음

이를 수학적 계산의 용이성을 위해 $\frac{1}{2}\|\vec{w}\|^2$ - (6) 문제로 변형

(6)의 식을 라그랑주 승수법(Lagrange Multiplier Method)을 적용하여 변형한다면

$$L(w, b, \alpha) = \frac{1}{2}\|\vec{w}\|^2 - \sum_{i=1}^n \alpha_i [y_i(\vec{w} \cdot \vec{x}_i + b) - 1]$$

Minimize w.r.t. \vec{w} and b maximize w.r.t. $\alpha_i \geq 0 \forall i$ - (7)

(7)의 식을 w와 b에 대해 미분하게 된다면

$$\frac{dL}{dw} = \sum_{i=1}^n \alpha_i y_i \vec{x}_i = \vec{w} - (8)$$

$$\frac{dL}{db} = \sum_{i=1}^n \alpha_i y_i = 0 - (9)$$

(8)과 (9)의 식을 (7)의 식에 대입하게 된다면,

$$L(\alpha) = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j \vec{x}_i^T \vec{x}_j$$

- (10) 이는 α 에 대한 maximization 문제로 정리됨.

KKT 조건에 의하면 α 의 값이 0이 아닌 경우 경계선을 정하는 Sample이 되므로 α 값을 통해 Support Vector를 구분할 수 있음. 또한 α 값을 구하게 된다면 (8)과 (9)를 통해 w와 b를 구할 수 있게 되므로 구분선 또한 구할 수 있음.

Soft한 linear SVM의 경우 (6)번 식의 $\frac{1}{2}\|\vec{w}\|^2$ 의 문제를

$$\begin{aligned} & \frac{1}{2} \|\vec{w}\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*) \\ \text{s.t. } & (\vec{w}^T \vec{x}_i + b) - y_i \leq \epsilon + \xi_i \\ & y_i - (\vec{w}^T \vec{x}_i + b) \leq \epsilon + \xi_i^* \\ & \xi_i \xi_i^* \geq 0 \end{aligned}$$

- (11) 로 변환한다.

위 식에서 ϵ 는 노이즈의 허용 범위이며 ξ 는 허용 범위를 넘어선 노이즈의 거리이다.

식에 라그랑주 승수법(Lagrange Multiplier Method)을 적용하여 변형한다면

$$\begin{aligned} L = & \frac{1}{2} \|\vec{w}\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*) - C \sum_{i=1}^n (\eta_i \xi_i + \eta_i^* \xi_i^*) - \sum_{i=1}^n \alpha_i (\epsilon + \xi_i + y_i - \vec{w}^T \cdot \vec{x}_i - \\ & b) - \sum_{i=1}^n \alpha_i^* (\epsilon + \xi_i^* - y_i + \vec{w}^T \cdot \vec{x}_i + b), \alpha_i^* \eta_i^* \geq 0 \end{aligned}$$

- (12)의 식을 앞서 한 것처럼 w와 b 그리고 ξ 에 대해 미분하게 된다면

$$\begin{aligned} \frac{dL}{dw} &= \sum_{i=1}^n (\alpha_i^* - \alpha_i) \vec{x}_i = \vec{w} \\ \frac{dL}{db} &= \sum_{i=1}^n (\alpha_i^* - \alpha_i) = 0 \\ \frac{dL}{d\xi_i^*} &= \alpha_i^* - \eta_i^* = C \end{aligned}$$

- (13)가 되며 이를 (12)식에 대입하면

$$\begin{aligned} L = & \frac{1}{2} \sum_{i,j=1}^n (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j) \vec{x}_i^T \vec{x}_j - \sum_{i,j=1}^n \alpha_i (\epsilon + \xi_i + y_i - \vec{w}^T \cdot \vec{x}_i - b) - \epsilon \sum_{i,j=1}^n (\alpha_i^* + \alpha_i) \\ & + y_i \sum_{i,j=1}^n (\alpha_i^* + \alpha_i) \end{aligned}$$

$$s.t. \sum_{i=1}^n (\alpha_i - \alpha_i^*) = 0, \alpha_i, \alpha_i^* \in [0, C]$$

- (14)로 α 에 대한 maximization 문제로 변형됨

KKT 조건에 의하면 α 또한 α^* 의 값이 0과 C 사이인 경우 경계선을 정하는 Sample이 되므로 α 값을 통해 Support Vector를 구분할 수 있음. 또한 α 값을 구하게 된다면 (8)과 (9)를 통해 w와 b를 구할 수 있게 되므로 구분선 또한 구할 수 있음.

Nonlinear한 SVM의 경우 동일한 과정을 적용하되 (\vec{x}_i, \vec{x}_j) 형태의 차원을 커널(Kernel)을 사용하여 $\phi(\vec{x}_i, \vec{x}_j)$ 형태의 차원으로 바꾸는 것이므로 상기 (10) 식과 (13)식의 $\vec{x}_i^T \vec{x}_j$ 부분을 $K(\vec{x}_i, \vec{x}_j)$ 형태의 함수로 변형해주기만 하면 됨. 자세한 내용의 경우 Appendix 3. 참고

커널(Kernel)

커널 트릭(Kernel trick)이란 수학적 기교를 통해 새로운 특성을 많이 만들지 않아도 고차원에서 분류기를 학습시키는 방법이다. 즉, 저차원 공간을 고차원 공간으로 매핑(mapping)시켜주는 과정을 의미하며 SVM을 적용하기 위해 커널(Kernel)을 반드시 거쳐야 한다. 고차원으로 표현할 때 요구되는 연산량이 크고, 데이터셋에 특성을 많이 추가하면 연산비용이 증가하기 때문에 커널 함수가 많이 사용된다. 이는 비선형의 단점을 보완할 수 있으며, 데이터를 확장하지 않고도 확장된 특성들에 대한 데이터 간 거리를 계산할 수 있으므로 데이터를 간단히 만들 수 있다. 커널함수를 이용하여 커널트릭을 가능하게 만드는데, 이 과정을 이해하기 위해 Lagrangian Dual Problem를 사용하겠다.

$$L_D = -\frac{1}{2} \sum_{i,j=1}^n (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j) x_i^T x_j - \epsilon \sum_{i,j=1}^n (\alpha_i^* + \alpha_i) + \sum_{i,j=1}^n y_i(\alpha_i^* - \alpha_i)$$

해당 식에서 $x_i^T x_j$ 에 커널함수를 사용하여 아래 식과 같이 $K(x_i x_j)$ 으로 표현하여 고차원 공간으로 변형해준다.

Dual Lagrangian problem with Kernel trick

$$L_D = -\frac{1}{2} \sum_{i,j=1}^n (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j) K(x_i x_j) - \epsilon \sum_{i,j=1}^n (\alpha_i^* + \alpha_i) + \sum_{i,j=1}^n y_i(\alpha_i^* - \alpha_i)$$

Decision function

$$\sum_{i=1}^n (\alpha_i^* + \alpha_i) \Phi(x_i) \rightarrow f(x) = \sum_{i,j=1}^n (\alpha_i^* - \alpha_i) K(x_i x_j) + b$$

결과적으로 간편하게 고차원공간으로 매핑하고 다음과 같이 비선형적인 회귀식을 도출할 수 있다. 커널트릭으로 활용할 수 있는 함수는 대표적으로 RBF, Linear, Polynomial이다.

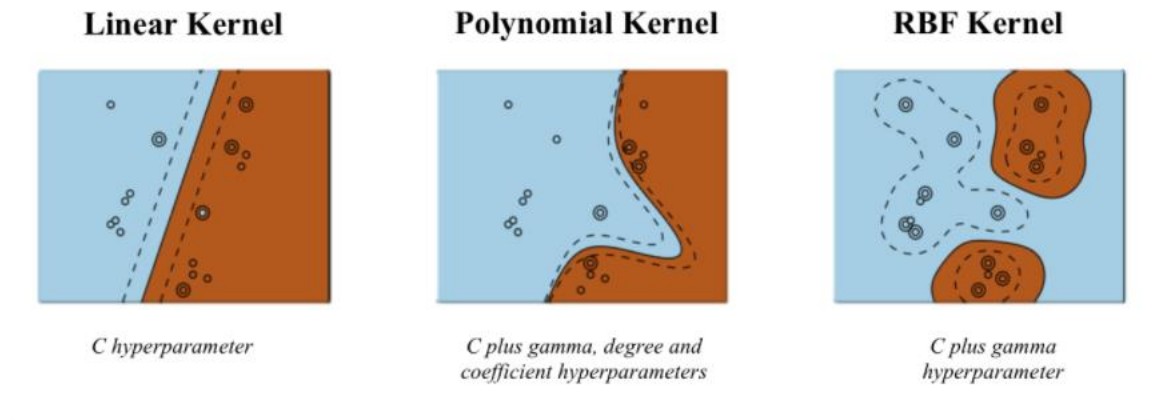


그림 5) 커널(Kernel)의 종류

연구 과정에 따라 값을 명확하게 산출해주는 함수를 선택해야 한다. 본 연구가 선택한 RBF(Radial Basis function)커널은 모든 차수의 모든 다항식을 고려한다. 이를 선택한 이유는 Class가 여러 개일 때 정확하게 분류해주기 때문이다. RBF커널함수는 다음과 같다.

$$e^{-\gamma(a-b)^2}$$

RBF커널은 차원이 무한한 특성 공간에 매핑할 수 있다. RBF가 무한공간에서 적용되는 방법은 새로운 데이터가 들어오면 가까운 데이터들의 영향을 받아 분류하는 것이다. γ 는 cross validation으로 결정되며, 두 점 a, b의 거리를 조절해주고 결론적으로 얼마나 영향을 줄 것인지 결정한다. 이러한 과정을 통해 데이터를 고차원 무한공간으로 매핑시킬 수 있는 것이다.