

LSTM을 이용한 주가 예측 모델 및 퀀트(Quant) 머신 개발

20928 정준영

내용

I. 서론	2
II. LSTM을 이용한 미래 주가 예측	3
1. 순환 신경망(RNN)	3
2. 장단기 메모리 신경망(LSTM)	5
3. 미래 주가 예측	7
III. 퀀트	10
1. 주식 투자의 기초	10
2. 투자 지표와 퀀트 투자 전략	12
IV. 적용	16
V. 결론	20

I. 서론

LSTM에 대해 알게 된 후 시계열 데이터¹의 대표적인 예시인 주식의 가격을 예측해보는 활동을 진행해보고 이를 실제 주식 투자에 활용해보고 싶어졌다. 이를 위해 LSTM과 yFinance 모듈을 사용하여 과거 주식데이터를 가져와 이를 학습하고 미래의 주식 가격을 예측해보는 파이썬 코드를 작성하였다. 그러나 가격 외에 여러 지표를 추가하고 하이퍼파라미터를 조정해 보아도 모델은 항상 미래의 주식 가격이 하락한다고 예측하여 원하는 결론에 도달하지 못하였다. 이를 해결하기 위해 [‘미국 주식으로 시작하는 슬기로운 퀀트투자, 김용환•Yubin Kim 저, 한빛미디어’](#)를 읽어보았고 여러 주식 투자에 관련된 개념들과 지표들을 배우고 finterstellar 라이브러리를 사용하여 예제를 따라해보며 퀀트 투자의 기초를 익혀보았다. 이를 이용하여 기존 목표였던 미래의 주식 가격 대신 자동으로 직접 지정한 조건에 따라 매도/매수를 하는 나만의 퀀트 머신을 개발해 보았다.

본 탐구에서는 위의 LSTM을 통해 미래 주식 가격 예측을 시도하고 실패한 경험과 책을 통해 공부하고 실습한 내용들 그리고 마지막으로 나만의 퀀트 머신을 개발하는 과정을 차례대로 설명하겠다.

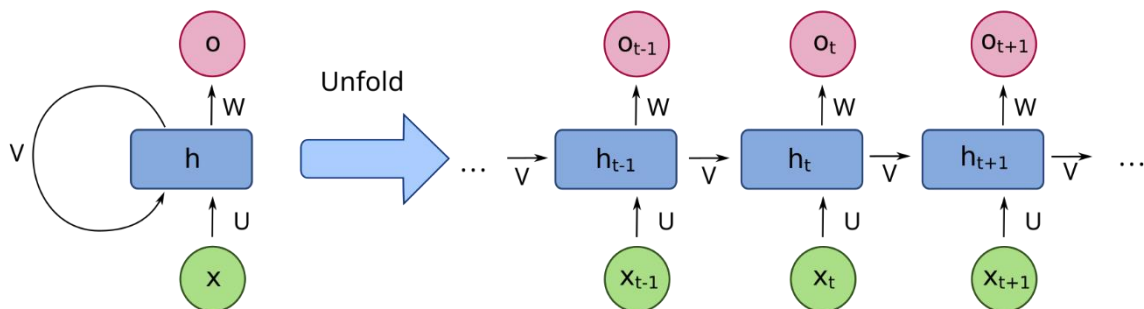
¹ 시간의 경과에 따라 일정 기간 동안 관측 값이 기록되어 있는 데이터로 데이터의 흐름의 변화 패턴을 파악함으로써 미래 예측이 가능함 ([블로그](#))

II. LSTM을 이용한 미래 주가 예측

주식 가격 예측에 앞서 RNN과 LSTM에 대해서 설명한 후 미래 주식의 가격 예측을 LSTM을 이용하여 수행하는 과정을 설명하겠다.

1. 순환 신경망(RNN)

순환 신경망은 이전 출력을 현재의 입력으로 받아 순차 데이터 또는 시계열 데이터(이러한 데이터를 시퀀스(Sequence)라 부른다)를 통해 데이터의 시간적 흐름을 학습하여 순차적 예측을 수행하는 심층 신경망이다. 앞의 과거 데이터를 입력으로 받으므로 순서가 있는 데이터에 특화되어 있으며 문맥을 파악해야 하는 자연어 처리(NLP) 모델이나 과거 기상 데이터를 이용하여 미래의 기상 상태를 예측하는 기상 모델, 기계 번역과 음성 인식 등에 사용된다. 다음으로는 RNN의 구조 및 원리에 대해 설명하겠다.^{2 3}



좌측은 RNN의 구조를 단순화한 것이다. x 는 입력층, h 는 은닉층, o 는 출력층이다. 기존 모델의 경우 입력층 \rightarrow 은닉층 \rightarrow 출력층으로 전달된다. 그러나 RNN에서는 은닉층이 현재의 입력층에 더해 과거의 은닉층으로부터 추가로 정보를 받게된다. 이를 그림에서는 은닉층 h 에서 순환하는 화살표를 통해 표현한 것이다. 이를 펼치면(unfold) 우측 그림이 된다. t 는 시간이며 h_t 를 현재라 했을 때 입력층 x_t 에 더해 과거 은닉층 h_{t-1} 의 정보도 제공받는다는 사실을

² [순환 신경망\(RNN\), wikidocs](#)

³ [RNN이란 무엇인가요?, IBM](#)

알 수 있다. 모든 정보는 시퀀스로 제공되므로 최종 출력을 제외한 모든 출력층에서는 시퀀스 데이터를 반환해야 한다. 기본적으로 위와 같은 형태를 띄지만 은닉-은닉 순환이 아닌 출력 정보가 대신 다음 은닉층에 제공되거나 출력 정보가 다음 출력층에 제공되는 출력-은닉 순환과 출력-출력 RNN 또한 존재한다. ^{4 5}

그러나 이러한 RNN에는 매우 큰 문제점이 존재한다. 만약 시퀀스 데이터가 매우 길다면 그래디언트⁶ 소실 또는 그래디언트 폭발 문제가 발생할 수 있다. 이는 학습 최적화를 위한 손실 함수(그림상의 W 에 해당)의 그래디언트를 계산할 때 발생한다. 이는 손실 함수를 계산하는 과정에 가중치를 시간이 지날 때마다 계속 곱해주는데 이 가중치가 너무 크거나 작을 때 손실 항이 매우 작아지거나 매우 커지면서 발생한다. 가중치 W 를 곱할 때 시퀀스가 너무 길면 기울기가 소실 또는 폭발한다. 시퀀스가 무한정 길 때 가중치 W 는 아무리 작아도 등비급수의 수렴/발산 조건에 의해 $|W| < 1$ 이라면 0에 수렴, $|W| > 1$ 이라면 발산하게 된다. 이를 기울기 소실/폭발이라고 부르는 것이다. 이를 해결하기 위해서는 $|W| = 1$ 이어야만 된다. 즉 시간이 길어질수록(= 시퀀스 데이터의 길이가 길수록) 해당 문제가 더 심하게 발생하게 된다. 이를 해결하기 위한 방법에는 1. 그래디언트 클래핑 / 2. TBPTT / 3. LSTM이 있다. 본 탐구에서는 LSTM만을 다루겠다. ⁷

⁴ [순환 신경망, 위키피디아](#)

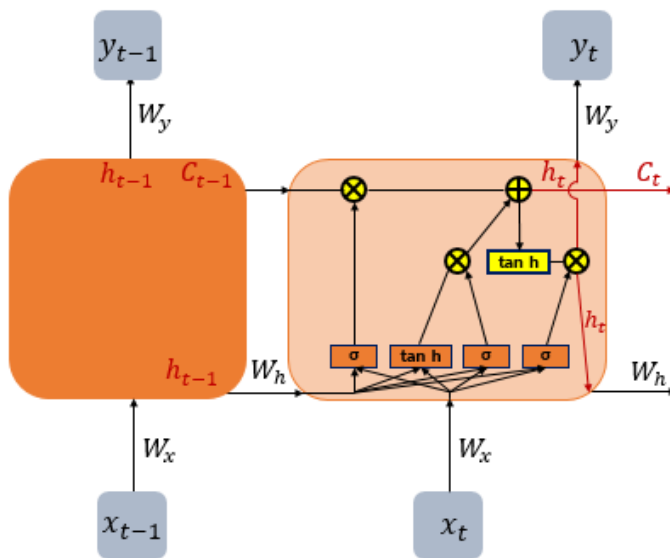
⁵ [머신 러닝 교과서 with 파이썬, 사이킷런, 텐서플로, 세바스찬 라시카, 바히드 미자리리, 길벗, P.668~80](#)

⁶ 그래디언트와 기울기는 1차원을 제외하고는 엄연히 다르지만 의미가 비슷하므로 그래디언트 대신 기울기를 넣어 기울기 손실/폭발 이라고 할 수도 있다

⁷ [기울기 소멸 문제, 위키피디아](#)

2. 장단기 메모리 신경망(LSTM)

LSTM은 기존 RNN의 기울기 소실 문제를 해결하기 위해 나왔다. 기존 RNN은 가중치를 곱하는 과정에 기울기가 소실될 뿐더러 시퀀스가 길게되면 RNN모델은 과거 데이터를 망각하게 되며 이를 장기 의존성 문제라 한다. 이러한 문제를 해결하기 위해 RNN모델의 기울기 소실 문제를 해결하며 과거 데이터를 지속적으로 기억하도록 만들어 줘야한다. 이를 위해 기존의 은닉층을 대체하는 메모리 셀(Memory Cell)이라는 새로운 요소를 삽입하였다.



다음 그림은 LSTM의 구조를 보여준다. 기존 은닉층이 메모리 셀로 대체되어 훨씬 복잡해졌다. 앞서 말했듯 $W = 1$ 을 유지하기 위해 각 메모리 셀에 순환 에지가 존재한다. 이 순환 에지의 출력을 셀 상태라 한다. 이전 셀 상태는 어떤 가중치와도 직접 곱해지지 않고 현재 타임에 대한 셀 상태를 얻게된다. 메모리 셀의 흐름은 몇 개의 게이트로 제어된다. x 는 입력층, C 는 셀 상태이다. 각 입력을 활성화 함수인 시그모이드 함수(σ)⁸와 하이퍼볼릭탄젠트(\tanh)⁹를 거쳐 일련된 가중치로 표시된다. 이러한 함수를 통과하는 유닛을 게이트라 부른다.

⁸ 입력된 데이터를 0과 1 사이의 값으로 출력하는 비선형 함수. $\frac{1}{1+e^{-t}}$ 로 정의됨 ([지식백과](#))

⁹ 치역이 -1에서 0인 함수로 시그모이드 함수보다 치역이 넓어 기울기 소실 문제가 비교적 적은 빈도로 발생함. $\frac{e^x - e^{-x}}{e^x + e^{-x}}$ 로 정의됨 ([블로그](#)) ([지식백과](#))

LSTM 셀에는 망각 게이트(Forget Gate), 입력 게이트(Input Gate), 출력 게이트(Output Gate)가 있다.

- 망각 게이트(ft)는 메모리 셀이 무한정 성장하지 않도록 셀 상태를 재설정한다. 다음 셀 상태로 넘어갈 때 통과시킬 과거 데이터와 억제할 과거 데이터를 결정한다.
- 입력 게이트(it)는 셀 상태를 업데이트 해준다.
- 출력 게이트(ot)는 은닉 유닛의 출력 값을 업데이트 해준다.

LSTM의 양방향 순환 버전인 bi-LSTM가 있고, GRU라는 LSTM보다 구조가 단순하고 계산이 효율적인 방식도 존재한다. ^{10 11}

¹⁰ [장단기 메모리, 위키피디아](#)

¹¹ 머신 러닝 교과서 with 파이썬, 사이킷런, 텐서플로

3. 미래 주가 예측

위에서 설명한 LSTM의 구조를 살펴보면 매우 복잡하다. 이러한 모델의 레이어 하나하나를 모두 구현하는 것은 매우 어려울 것이다. 따라서 데이터 전처리와 관련된 것은 머신러닝 라이브러리인 사이킷런(Scikit-learn)¹²을 사용하였고, LSTM 모델을 정의하고 학습 및 예측은 Keras¹³ 라이브러리를 사용하였다. 우선 주식 데이터를 수집하여야 한다. 이를 위해 [Yahoo Finance](#)¹⁴의 데이터를 사용하였다. 데이터를 가져오는데 사용한 모듈은 yfinance이다. 주식 종목은 다음과 같이 설정했다.

```
# 주식 티커 및 회사명
tickers = ['AAPL', 'MSFT', 'T', 'NVDA', 'GOOGL']
names = ['애플', '마이크로소프트', '테슬라', '엔비디아', '구글']
```

애플, 마이크로소프트, 구글의 경우 MAGA¹⁵에 해당되는 거대 기업이며 테슬라, 엔비디아의 경우 버블이 많이 끼었다고 평가되거나 현재 주가가 급성장 중인 종목들이다. 구글(실제로 구글은 주식회사가 아니며 Alphabet Inc.의 자회사이다. 따라서 회사명도 알파벳이 맞으나 편의상 구글이라 칭한다)의 경우 티커가 GOOG와 GOOGL이 존재하는데 각각 의결권이 없는 Class C와 의결권이 존재하는 Class A이다. 가격은 의결권이 없는 Class C가 살짝 더 높다. 날짜는 2018-01-01 ~ 2023-12-31이다.

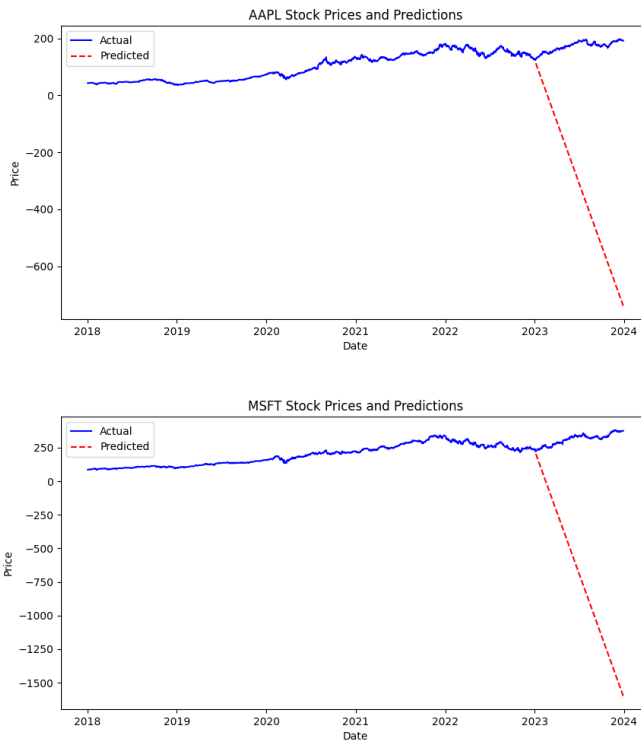
¹² 사이킷런은 머신러닝을 위한 라이브러리지만 여기서는 데이터를 전처리하기 위해 사용한다. 파이토치, 텐서플로 등 다른 라이브러리도 존재하지만 이들은 딥러닝 라이브러리로 사이킷런에 비해 무겁고 느리다

¹³ Keras는 구글에서 개발한 라이브러리로 Tensorflow(외에도 Pytorch등으로 변경 가능함) 위에서 작동한다. Keras는 딥러닝을 쉽게 구현할 수 있도록 유저 친화적이며 간단한 코드로 심층 신경망을 구현 가능하다. Tensorflow보다 확장성이 떨어져 세부 조작이 불가하지만 사용하기 쉽기 때문에 자주 사용된다 ([티스토리](#))

¹⁴ 어째서인지 학교 와이파이로 접속하면 '유해사이트'라 뜨고 차단된다. 이는 yfinance에서 데이터를 가져올 때도 마찬가지로 학교에서 코드를 실행할 때 VPN을 켜 상태로 해야했다...

¹⁵ Microsoft, Apple, Google, Amazon을 이르는 말로 세계 최고의 시가 총액을 자랑하는 기업들이다

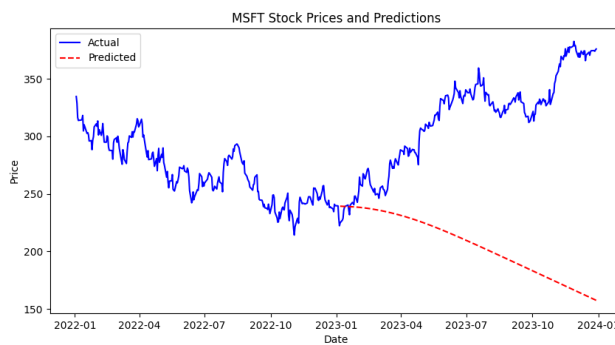
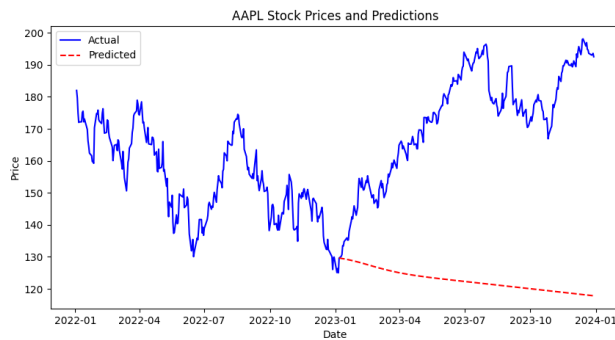
처음에는 단순히 가격만을 제공한 다음 LSTM으로 학습해보았다. 모델의 성능을 확인하기 위함이므로 애플과 마이크로소프트의 주식만 우선 학습하고 예측해보았다. 가격은 종가를 기준으로 학습하였다. 학습 데이터의 날짜는 2018-01-01 ~ 2022-12-31이며 2023-01-01부터 2023-12-31까지의 가격을 예측해 볼 것이다.



심각한 결과가 나왔다. 애플과 마이크로소프트 두 종목 모두 상장폐지를 넘어 바닥을 한참 뚫었다. 만약 파생상품(옵션, 선물)을 매도한 사람이라면 파산했을 것이다. 반면 풋옵션을 매수한 사람은 엄청난 부자가 되었을 것이다. 현실적으로 불가능하며 실제 일어나서도 안되는 이러한 예측 그래프가 나오는 문제점을 해결하기 위해 책과 인터넷을 찾아보며 해결책을 찾아보았다. LSTM의 경우 주식 가격만으로 학습하게 된다면 가장 손실이 적게 학습된 모델은 전날 가격과 금일 가격이 같다고 예측하게 된다. 또한 주식의 가격은 랜덤성이 짙으며 가격뿐만 아니라 다른 지표들도 학습에 사용되어야 하며, 가격이 아닌 가격의 변화율 등 날짜에 따라 얼마나 변화하는지를 학습해야 된다.¹⁶

¹⁶ <https://codingapple.com/unit/deep-learning-stock-price-ai/>

이를 참고하여 단순 가격만이 아닌 증가 변화율(현재 증가 - 전날 증가), 거래량 변화율(현재 거래량 - 전날 거래량), 5일/20일 이동평균을 모두 사용하여 학습해보았다.



상장 폐지는 막았다. 그러나 여전히 실제 주식은 가격이 점점 상승하는 것과 반대로 예측된 가격은 점점 하락하고 있다. 이러한 문제들을 해결하기 위해 주식에 대한 기초적인 개념들과 주식 예측위해 더 자세하고 전문적인 지식이 필요함을 느꼈다.

III. 퀀트

주식의 미래 가격을 예측하는 것은 결국 미래의 주식이 상승할지 하락할지 예측하는 것이지 미래의 가격이 정확히 얼마인지 알 필요가 없다. 그렇다면 주식이 미래에 상승할지 하락할지 어떻게 알 수 있을까? 그에 대한 답을 찾기 위해 [‘미국 주식으로 시작하는 슬기로운 퀀트투자, 김용환•Yubin Kim 저, 한빛미디어’](#)를 읽어보았다.

1. 주식 투자의 기초

경영에 관심 없는 일반인들이 주식 투자를 하는 이유는 당연히 돈일 것이다. 그러나 투자를 하려면 우선 어떻게 해야 하는지를 알아야 한다. 워런버핏을 비롯한 많은 전문가들은 장기투자를 권장한다. 당연하게도 통화량이 증가하므로 물가는 상승하며 주식가격도 상승하게 되므로 장기적으로 보면 물가, 부동산, 주식 모두 상승하며 반대로 화폐가치는 계속 하락하게 될 것이다. 이러한 관점에서 보면 절대 근수십년안에 망할리가 없는 다국적기업에 투자를 한다면 거의 확실하게 투자는 성공하게된다. 그러나 이러한 장기투자는 시간이 매우 오래 걸리며 대부분은 수익률이 그렇게 높지 않다. 그저 현금을 가지고 있는 것보다 살짝 나은 수준에 그칠지도 모른다. 그렇다고 무작정 단기투자를 한다면 실패할 확률이 매우 높다. 단기투자를 위해서는 여러 지표들과 기업의 가치를 분석하고 적절하게 주식을 매수하거나 매도하여 수익을 얻어야 한다.

자본주의 시장에서 자본은 일부 예외를 제외하면 항상 수요-공급의 원칙에 따른다. 주식시장의 수요는 매수 주문, 공급은 매도 주문이 될것이다. 사람들은 주식을 보유함으로써 이익이 생길것이라 생각되면 매수를 할것이다. 대부분 기업의 가치가 상승할 때(예를들어 애플이 아이폰을 새로 출시하여 별 논란 없을 때) 투자자들은 해당 기업이 더 큰 수익을 가져다 줄것이라 생각하며 해당 기업의 주식을 매수할 것이다. 그렇다면 당연히 수요가 증가하므로 주식의 가격은 상승한다. 반대로 기업의 중대한 논란이 발생하였을 때(예를들어 애플의 배터리 게이트) 기업의 가치는 하락할것이고 당연히 주주들은 주식이 하락할것이라 예측하고 매도를 할것이다. 당연하게도 그 결과는 주식 가격의 하락으로 이어진다. 언뜻 보면 주식의 가격은 기업의 가치에 따라가는 것 처럼

보이나 실은 그렇지 않다. 기업의 가치가 주가에 많은 영향을 끼치는 것은 사실이나 결국 주가를 결정하는 것은 투자자들의 매수/매도 주문이다. 기업이 아무리 가치가 높아도 매수하는 투자자가 적으면 가격은 낮을 수 있으며, 실제 가치는 낮으나 매수자가 많으면 높은 가격대를 형성할 수도 있다. 이러한 사례로는 파산 보호 신청을 한 렌터카 회사 Hertz가 갑자기 주가가 급상승 하거나, 분명 기업의 가치는 저만큼이 아닌데도 불구하고 경제학적으로 설명이 어려운 주가를 형성중인 테슬라가 잘 보여주고 있다. 이들 모두 투자자들은 기업의 가치보다는 그저 주식의 가격이 올라갈 것 같으니 매수를 하는것이다. 따라서 기업의 장기적인 가치를 분석하여 저평가된 가치주를 찾아내어 가치투자를 하는 방법도 있지만, 기업의 가치와는 상관없이 사람들이 주식을 얼마나 매수/매도 할지 예측하여 단기적으로 매수하였다가 이익을 본 다음 바로 매도해버리는 방법도 있다. 기업의 가치를 분석하는 것은 전문 직업인 애널리스트가 하는 일이다. 이러한 일을 지식도 없는 일반인이 제대로 할 리가 없다. 따라서 단순히 사람들이 주식을 얼마나 매수/매도하는지 그 추세만을 파악하여 투자를 진행하는 단기투자 방법을 사용할 수 밖에 없다(물론 앞서 말했듯 장기투자도 되나 단시간내 높은 수익을 원한다면 장기투자는 불가능하다). 기업의 가치를 분석하는 방법은 재무제표를 분석하여 해당 기업의 가치를 파악 할 수 있다. 반면 투자자들의 심리와 추세를 파악하기 위해서는 여러 지표들을 분석하여야 한다. 이러한 지표들을 컴퓨터가 자동으로 분석하여 매도/매수 주문을 하는 투자 방법을 바로 퀀트라 부르는 것이다.

2. 투자 지표와 퀀트 투자 전략

투자 지표는 무수히 많다. 그 중에서 몇 개만 골라 설명하고 실습해 보겠다. 이를 위해 우선 finterstellar(핀터스텔라) 라이브러리를 사용한다. 해당 라이브러리는 쉽게 주식 시세를 가져오는 것부터 여러 지표들을 계산하고 주가를 분석하거나 백테스팅¹⁷ 을 쉽게 할 수 있도록 미리 모든 코드들이 짜여진 라이브러리다. 해당 라이브러리에 존재하는 함수만 가져와서 사용하면 바로 손쉽게 계산, 백테스팅, 전략 평가가 가능하다. 다음은 백테스팅 결과를 읽고 평가하기 위한 용어 정리이다.

- CAGR(연평균수익률): 투자기간을 1년 단위로 환산하여 연 평균 수익이 얼마인지 특정하는 지표. 1년보다 짧으면 단리, 길면 복리로 계산함
- Accumulated Return(누적수익률): 전체 투자기간동안 누적된 수익률
- Average Return(평균수익률): 트레이딩 동안 한 사이클(매수부터 매도까지) 돌렸을 때 발생한 사이클당 수익들의 평균
- Benchmark Return(벤치마크수익률): 개별 종목 투자에서 해당 종목을 전략을 이용하지 않고 같은 기간동안 계속 보유하고 있었을 때의 누적 수익률
- Number of Trades(거래횟수): 전체 투자기간동안 발생한 총 사이클 수
- Number of Win(성공횟수): 전략이 성공한 즉 수익이 +인 사이클의 수
- Hit Ratio(성공확률): 전략이 성공한 확률로 (성공횟수)/(거래횟수)로 계산
- Investment Period(투자기간): 테스트를 진행한 총 투자기간
- Sharpe Ratio(샤프비율): 위험 대비 수익률.
$$\text{샤프비율} = (\text{수익률} - \text{무위험이자율}) / (\text{수익률} - \text{무위험이자율}) \text{의 표준편차}$$
- MDD(최대낙폭): 투자기간 중 투자자가 입을 수 있는 최대 손실률

¹⁷ 설정한 투자 전략에 따라 트레이딩을 하였을때 성과가 어떨지 시뮬레이션 해보는 것

다음으로는 여러 지표들과 이를 이용한 투자 전략 및 결과를 일부 소개하겠습니다.
책의 실습코드는 https://github.com/finterstellar/quant_machine 에 있다.

- 모멘텀: 모멘텀 전략은 이미 상승세인 주식을 매수하여 더 오르기를 기대하는 전략이다. 주로 높은 수익을 원하는 투자자들이 사용하지만 고점에서 매수하여 최고점에서 매도해야하는 전략이므로 리스크가 크다.
- 평균회귀: 평균 회귀 전략은 평소보다 주가가 많이 하락한 종목을 매수하여 다시 원래 주가로 회복하기를 기대하는 전략이다. 수익률은 낮지만 투자지표라는 확실한 수치적 시그널이 존재하여 저평가 되고있는 종목만 잘 찾으면 안정적으로 수익을 올릴 수 있다. 이는 안정적인 주식이 박스형식으로 지그재그로 움직이는 것을 이용하여 박스의 하단부에 있을 때 매수 하여 박스의 상단부에 도달하면 매도하는 방식이다. 애초부터 안정적인(저점과 고점의 차이가 별로 없는) 종목에서 저점에서 매수하여 고점에서 매도하므로 수익은 낮지만 리스크가 적다.
- 이동평균: 일반적인 평균과는 다르게 이동 평균은 기준을 이동시키며 기준에서 일정한 범위만큼 떨어진 범위 내 수치만 평균내는 것을 말한다. 주가는 매 순간 변하므로 오늘 구하는 일주일 평균과 내일 구하는 일주일 평균 주가는 다르다. 따라서 기준에 따른 날짜 범위를 지정하고 그 범위 내에서 평균을 구하는 방식을 이용한다. 이동평균은 MA로 표현하며 기준으로부터 5일간의 주가의 평균이면 MA(5), 20일이면 MA(20)으로 표현한다. 이러한 각 날짜에 대한 이동평균을 이은 선이 이동평균선이다. 주식시장에서는 영업일만 계산하므로 중간에 공휴일이 있다면 뛰어넘는다. 이동평균은 기준날의 그 전 일정 날짜까지의 평균이므로 그 날짜 범위만큼의 이전 데이터가 존재하지 않으면 구할 수 없다. 예를들어 1월 1일부터 5일까지의 데이터 밖에 존재하지 않는다면 MA(3)을 구할 때 1월 2일까지는 구할 수 없고 3일부터 구할 수 있다. 최근 주가가 상승하는지 하락하는지 추세를 보려면 날짜 범위가 작은 이동평균선이 범위가 넓은 이동평균선 위에 있다면 최근 주가가 상승 중이라는 의미가 된다.

- MACD(이동평균수렴확산지수): 이동평균선은 과거 특정시점부터 지금까지의 평균이라 이를 이용하여 주가 추이를 관찰하다 보면 실제 추이에 비해 늦어질 수 밖에 없다. 이러한 문제를 해결하기 위해 최근 주가에 더 많은 가중치를 곱하여 평균내는 방법이 MACD이다. MACD를 사용하면 매매 전략은 MACD가 +면 매수(최근 주가가 상승 중 이라는 의미이므로), -면 매도한다. 이를 마이크로소프트 주가에 적용하면 MACD가 20년 2~3월 코로나 시기를 제외하고는 양수에 위치하는 것을 알 수 있다. 즉 마이크로소프트는 코로나 시기에 잠깐 하락했다가 이후에는 전체적으로 꾸준히 상승중이라 보면 된다. MACD가 양수일 때 매수이므로 해당 시그널을 지정하여 백테스팅을 진행해보면 CAGR이 23.49%가 나온다. 그러나 벤치마크 수익률이 38.48%로 전략을 사용하지 않고 가만이 나뒹을 때가 더 수익이 높다. 다만 MMD를 보면 전략을 사용했을때가 -11.28%, 벤치마크 MMD는 -28.24%로 전략을 사용한 것이 안한것에 비해 리스크 리 적었다는 것을 알 수 있다. 이렇듯 주식 투자는 수익만 고려해서는 안되며 리스크 관리도 철저히 해야한다.
- RSI(상대강도지수): 주가 상승의 강도를 측정하는 지표. 일정 기간동안 일간 상승폭과 하락폭을 계산하고 그 중 상승폭이 얼마나 큰지 비율로 계산한다. 이 지표를 만든 웰레스 와일더는 RSI가 70이상이면 과매수(주식을 너무 많아 사서 과열된 상태), 30이하면 과매도 구간으로 분류했다. 저점에 사서 고점에 팔아야 돈을 벌 수 있으니 과매도 구간(공급이 수요보다 높으므로 가격이 하락)에 매수를 하고 과매수 구간(수요가 공급보다 높으므로 가격이 상승)에서 매도해야한다. 이를 미국의 통신사 버라이즌(VZ)에서 백테스팅 해보면 평균회귀전략에서는 RSI가 30이하면 매수, 70이상이면 매도하면 CAGR이 7.2%이며, 모멘텀전략에서는 RSI가 70이상에서 매수, 50이하에서 매도하여 CAGR이 -2.94%로 손해를 봤다. 주식마다 평균회귀 또는 모멘텀 전략에 따라 성과가 달라지며 꾸준히 우상향하는 주식에는 모멘텀 전략을, 박스권에서 안정적으로 움직이는 주식은 평균회귀 전략을 사용하는 것이 더 성과가 좋다. 버라이즌의 2020년 주가는 저점이 50, 고점이 60으로 박스권에서 안정적으로 움직이므로 평균회귀전략이 더 나으며 실제 결과도 그렇게 나온다.

- Envelope(엔벨로프): 이동평균선을 중심으로 일정 범위만큼 밴드로 표시하는 지표. 밴드 안팎으로 총 4개의 영역이 발생하게 된다.



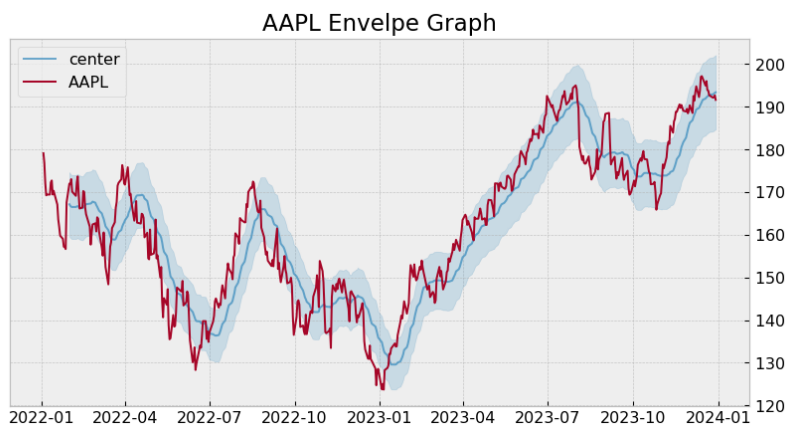
이를 구간 A, B, C, D로 표현해 보겠다. A는 주가가 밴드 상단부 밖에 있는 구간, B는 밴드의 중심(이동평균선)과 밴드의 상단, C는 밴드의 중심과 하단 사이, D는 밴드 하단부 밖 구간이다. 모멘텀 전략에서는 상승세에 매수하므로 A구역에서 매수하며 상승세가 꺾일 때 매도하므로 B구역에서 매도한다. 평균회귀 전략에서는 평소보다 낮을때인 D구간에서 매수하여 원래 주가인 중심선에 가까워지는 C구간에서 매도하게 된다. 이를 보잉(BA) 종목으로 백테스팅 해보자. 모멘텀 전략에서 CAGR이 38.38%, 평균회귀 전략에서 CAGR이 -22.11%, 벤치마크 수익률이 -33.40%이다. 보잉은 성장주가 아님에도 불구하고 모멘텀 전략에서 훨씬 우수한 성능을 보여준 것을 확인할 수 있다.

이 외의 변동성을 이용한 투자 지표들과 전략은 생략하겠다.

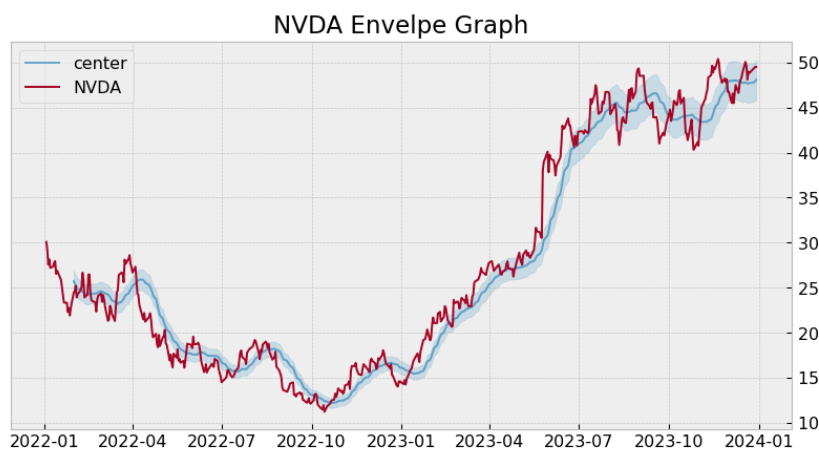
IV. 적용

이제 위에서 학습한 내용들과 실습 코드들을 사용하여 처음의 목표했던 5개의 종목에 대한 주가 상승/하락 예측 및 시그널을 이용한 퀀트 시뮬레이션을 진행해 보겠다. 책에서는 다양한 지표를 이용하여 자동으로 종목을 선정하는 완벽한 퀀트 머신을 만들었지만 본 탐구에서는 이미 5개의 종목이 선정되어 있으므로 각 종목을 엔벨로프 지수를 이용하여 각각 모멘텀과 평균회귀 전략을 이용하여 백테스팅을 수행하고 이를 통해 5개의 종목에 맞는 전략을 채택하여 자동으로 매수/매도하는 간단한 형태의 퀀트 머신을 만들어 보겠다.

엔벨로프의 밴드는 이동평균선의 상하 4.5%로 지정하였다. 이동평균은 MA(20)을 사용하였다.



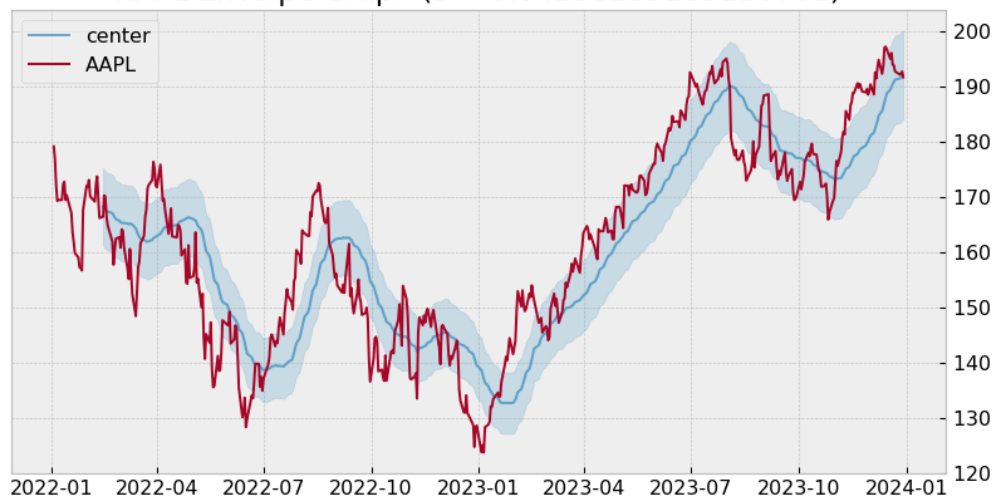
애플의 엔벨로프 그래프이다. 주가와 엔벨로프 밴드가 거의 비슷한 추이로 움직이고 있다.



엔비디아의 그래프이다. 분명 똑같이 4.5%로 설정했으나 보이는 두께는 애

플과 많이 다르다. 이는 애플과 엔비디아의 주식 변동률의 차이이다. 애플은 저점과 고점의 차이가 70%가 안된다. 그러나 엔비디아의 경우 약 500%의 차이가 난다. 이를 해결하기 위해 밴드 범위를 유동적으로 설정해줬고 가격 변동에 덜 민감해 지도록 이동 평균의 날짜를 30일로 변경해줬다. 밴드의 범위는 주가의 최고가와 최저가의 차이의 비율을 사용하여 $(0.033 + 0.015 * \text{차이비율})$ 로 설정해주었다. 여기서 $0.01 = 1\%$ 이다. 이후 다시 그래프를 생성해 보았다.

AAPL Envelope Graph (s = 0.04190155189136761)



애플의 엔벨로프 그래프이다. 이전보다 엔벨로프가 완만해졌으며 두께는 얇아졌다.

NVDA Envelope Graph (s = 0.08536631016042781)



엔비디아의 그래프는 이전보다 두께가 두꺼워졌다.

이제 해당 엔벨로프들을 이용하여 각 종목별로 모멘텀과 평균 회귀 전략을 이용하여 각각 백테스팅을 수행하여 비교해 보겠다. 백테스팅을 위해 매매에

드는 비용(즉 수수료)는 1%로, 가만히 은행에 저축할 시 얻는 수익인 무위험 이자율은 미국의 2022년 기준 금리(0.5%~4%)를 평균내어 1.75%로 설정하였다. 결과는 '[quant_res.txt](#)'에 저장되어 있다.

대부분의 종목과 전략에서 수익은 발생했으나 벤치마크 수익엔 못미쳤다. 다만 대부분에서 MDD가 전략을 사용했을때가 벤치마킹 MDD보다 나은걸 보아 적어도 리스크 측면에서는 괜찮은 결과를 얻었다 볼 수 있다. 다음은 벤치마킹 수익보다 더 나은 수익을 얻은 케이스들이다.

T 평균회귀 전략 결과:

CAGR: 3.30%

Accumulated return: 6.27%

Average return: 1.54%

Benchmark return : 3.13%

Number of trades: 6

Number of win: 5

Hit ratio: 83.33%

Investment period: 1.9yrs

Sharpe ratio: 0.07

MDD: -21.28%

Benchmark MDD: -32.00%

GOOGL 평균회귀 전략 결과:

CAGR: 11.63%

Accumulated return: 22.85%

Average return: 2.24%

Benchmark return : 3.07%

Number of trades: 10

Number of win: 8

Hit ratio: 80.00%

Investment period: 1.9yrs

Sharpe ratio: 0.38

MDD: -19.91%

Benchmark MDD: -41.65%

애플을 제외하고는 모든 모멘텀 전략에서 수익이 마이너스이다. 또한 모든 케이스에서 평균회귀 투자가 모멘텀 투자보다 좋은 수익을 보여줬다. 엔비디아를 제외한 종목들은 그래프의 모양이 안정적이라 평균 회귀가 강세를 보여줄것이라 생각했고 실제로도 그랬다. 또한 의외로 2022년 당시 성장주였던 엔비디아도 평균 회귀가 수익이 더 좋다는 사실이 신기하다.

V. 결론

주식의 가격을 예측해보기 위해 RNN과 LSTM의 원리에 대해 이해한 뒤 LSTM을 이용하여 주가 데이터를 학습하여 미래 1년간의 주가를 예측해 보았다. 결과가 심히 좋지 못하여 해결 방안을 탐색하고 가격이 아닌 가격 변화율과 다른 데이터들도 함께 학습시켰으나 여전히 좋지 못한 성능을 보고 좀 더 전문적인 지식을 배우면 해결될지 모른다는 생각에 책을 읽고 주식 투자와 퀀트의 기초 개념에 대해 공부를 하였다. 이후 해당 책에서 여러 지표들과 핀터스텔라 라이브러리 사용법을 익히고 실습해보았다. 학습한 내용을 바탕으로 처음 목표였던 5개의 종목에 대한 퀀트 머신을 만들어 보았다. 엔벨로프를 이용하여 주식의 가격을 이동평균선과 밴드를 통해 나타내고 가격과 엔벨로프 밴드간의 관계를 이용하여 각각 모멘텀 전략과 평균회귀 전략을 이용하여 5개 종목에 대한 백테스팅을 진행하였다. 결과는 벤치마킹 수익에 비해 좋지 못하나 벤치마킹 MDD보다 수치가 나았으며 처음 했던 LSTM을 이용한 주가 예측 결과에 비하면 적어도 수익은 대부분 발생하였으니 훨씬 나은 결과였다고 생각한다.

본 탐구에 이어 내년에는 본 탐구에서 언급하지 않은 다른 지표들과 종목을 여러 재무재표와 지표들을 이용하여 선정하는 방법을 학습하고 이를 이용하여 자동으로 종목 선정부터 매도/매수까지 수행하는 완벽한 퀀트 머신을 개발하고, 이를 실제로 사용하여 주식 투자를 해보고 싶다. 또한 여기에 더해 뉴스와 SNS의 언급과 반응을 수집하여 주주들의 심리를 반영하는 'センチメント 퀀트'를 개발해보고 싶다. 또한 성과가 좋다면 퀀트 머신을 배포하여 여러 사람들이 사용할 수 있도록 만들어보고 싶다.

본 탐구 'II. LSTM을 이용한 미래 주가 예측'과 'IV. 적용'에 사용된 코드들과 결과물 등은 모두 [깃허브](https://github.com/jjy0809/quant/)¹⁸에 업로드 되어져 있다.

¹⁸ <https://github.com/jjy0809/quant/>