Contents

Chapter 1

Introduction

1.1 Background

Stocks are equity investments that represent legal ownership in a company. Investing in stocks can be an efficient way to build wealth over time. Learning how to invest wisely and patiently over a lifetime can yield returns that far outpace the most modest income. Technical analysis is sometimes used in financial markets to assist traders make buying and selling decisions.

In order to better profit in the stock market, it is important to use technical analysis of financial markets, which can be used to help traders make good buying and selling decisions. In this study, we collected stock data from different sectors and tried to find ways to maximize or minimize profits by studying historical data and using technical tools.

1.2 Problem Statement

In summary, the goals of our study are twofold. The first one is to find strategies which can maximize or minimize profits and the second one is to find out the events that might affect the stock price automatically. To achieve the two goals, the tasks involved are as following:

1. Collect 50 stocks data from five years ago, including names of companies, dates, their prices (open, high, low, close) and trading volume.

2. Categorize companies based on their sectors and then use average daily log-return to find the most profitable companies in each sector.

3. Plot the trend of stock prices of different companies during the last three years sector by sector and describe their trends based on sectors.

4. Use two different models to develop trading strategies to make money from stocks, compare and evaluate the two strategies.

5. Crawl the financial news of Wall Street Journal from 2020-03-17 to 2020-03-24 and calculate word frequency and plot word cloud. Then find out events that affect the stock prices based on the results.

Chapter 2

Data Analysis

2.1 Collect data

We collect 50 stocks randomly from SP 500 Component Stocks from five years ago, including names of companies, dates, their prices (open, high, low, close) and trading volume. The specific time period is from 2017-10-01 to 2022-11-27. The module we used for collection is yfinance, which is an open-source tool that uses Yahoo's available APIs.

2.2 Categorize Companies

Based on 50 company stocks, selected randomly from 5 sectors in S & P 500 list (Crawling from **Wikipedia**), we category and split them accordin to their sectors, which are "Industrial", "Health Care", "Information Technology", "Communication Services", "Consumer Staples".
Next, to find the most profitable companies in each sector, we calculate the daily log-return for each company,denoted by r_i :

$$r_i = ln(\frac{P_i}{P_{i-1}}) \tag{2.2.1}$$

where P_i is the i-th day's stock price.
And add the values into a new column of this dataset. Since there are 252 trading days in a year, the average daily log-return of each company is calculated, denoted by $\overline{r}$ multiplied by 252 as the average annual log-return, denoted by R. i.e.,

$$\mu = \overline{r} * 252 \tag{2.2.2}$$

After doing this, we find the most profitable companies in each sector as the following table shows:

Sector	**The most profitable company**
Industrial	Cintas Corporation
Health Care	Agilent Technologies
Information Technology	Apple
Communication Services	Alphabet Inc. (Class C)
Consumer Staples	Archer Daniels Midland

Table 2.2.1: The most profitable company in each sector

2.3 Trend of Stock Price

The slope of a trend indicates how much the price should move each day. Steep lines, moving either upward or downward, indicate a certain trend. In this section, we plot the trend of all 50 stocks' daily closing price based on their sectors.
In the overall view, almost all the stocks experience a huge drop during March 2020 and some of them then experienced a upward trend.

Industrial

The figure below shows the trend of stock prices in the Industrial sector.

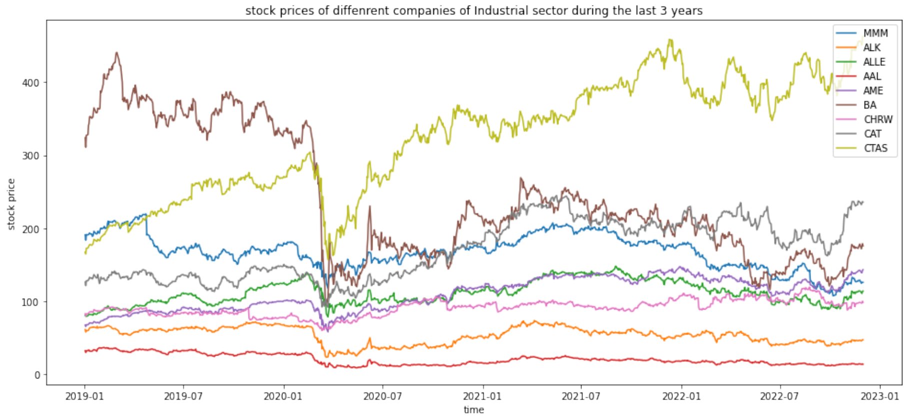


Figure 2.3.1: The trend of stock prices in the Industrial sector.

In an overall view, almost all stocks are trending down in March 2020. Stocks like AAL, ALK and CHRW are moving smoothly, with no huge fluctuations over three years, and some stocks show clear upward trends after the meltdown like MMM and CAT, which also coincided with the bull market in US stocks.
Besides, there are two stocks (BA and CTAS) which show great volatility. The stock BA began a downward trend after the 2019 plane crash and experienced severe losses during the epidemic. Although it rose a bit during the 2020 bull market, it still has not returned to pre-epidemic levels. The stock CTAS also experienced a huge drop during March 2020, but it quickly recovered and maintain the upward trend just as it did before the epidemic.

Health Care

The figure2.3.2 shows the trend of stock prices in the Health Care sector.
There are still some stocks always in a flat trend but more stocks show clear volatility in this sector. For example, stocks like BSX and CAH seem to have been oscillating at around $50, but stocks like A and BIIB experienced dramatic price fluctuations. What's more, an interesting phenomenon is that BIIB showed a huge drop around April 2019 and experienced a big rise and fall in mid-2021, which is quite different from others in this sector.

IT

Figure2.3.3 shows the trend of stock prices in the Information Technology sector.
From the figure, we can find that many stocks in this sector move very similarly. To be more specific, more than half of stocks show upward trends in the bull market after the meltdown, and most of them also experienced clear downward trends after December 2021. Beside, there are still some stocks which are always in flat trends during pass 3 years.

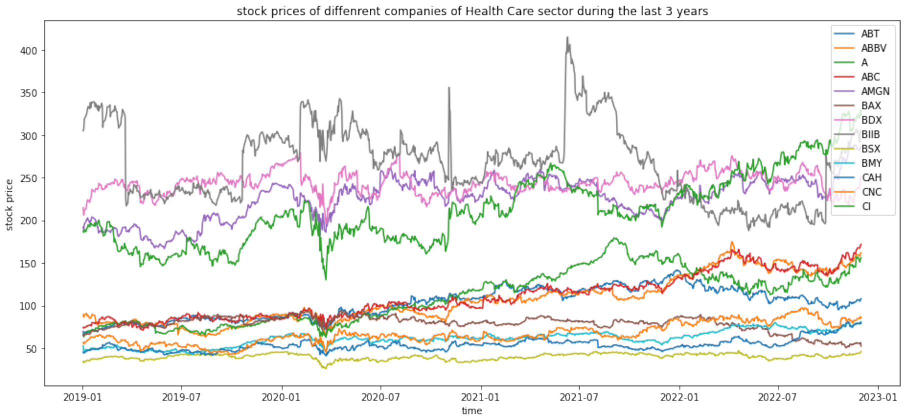


Figure 2.3.2: The trend of stock prices in the Health Care sector.

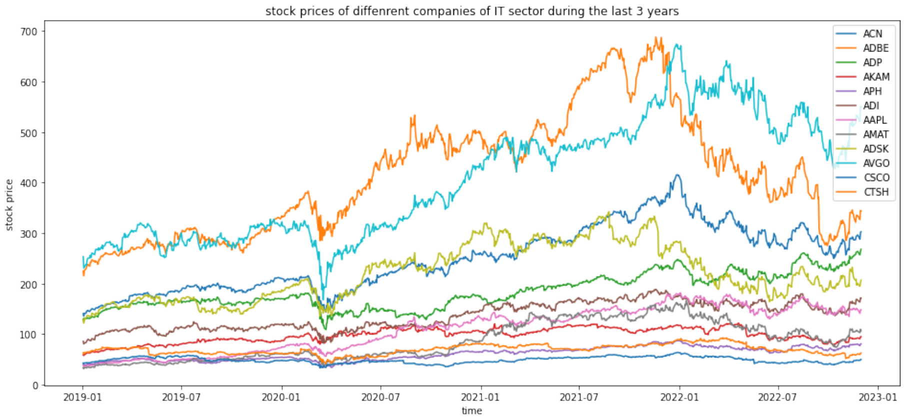


Figure 2.3.3: The trend of stock prices in the Information Technology sector.

Communication Services

The figure2.3.4 shows the trend of stock prices in the Commnication Service sector.
Except for CHTR, all stocks are in a relatively flat trend without much volatility. However, CHTR was in a continuous rise before the epidemic and after a decline in March 2020, it recovered quickly and continued to rise until October 2021. After October 2021, it experienced a huge drop till now.

Consumer Staples

Figure2.3.5 shows the trend of stock prices in the Consumer Staples sector.
In this sector, stocks except for CLX and STZ all have share prices below $100 and have a relatively flat overall trend. STZ has been on a steady upward trend, except for a huge drop in March 2020. The trend of CLX is different from any other stocks, as it seems to be unaffected by the March 2020 stock market crash. It continued to rise before the first half of 2020 and then dropped after that.

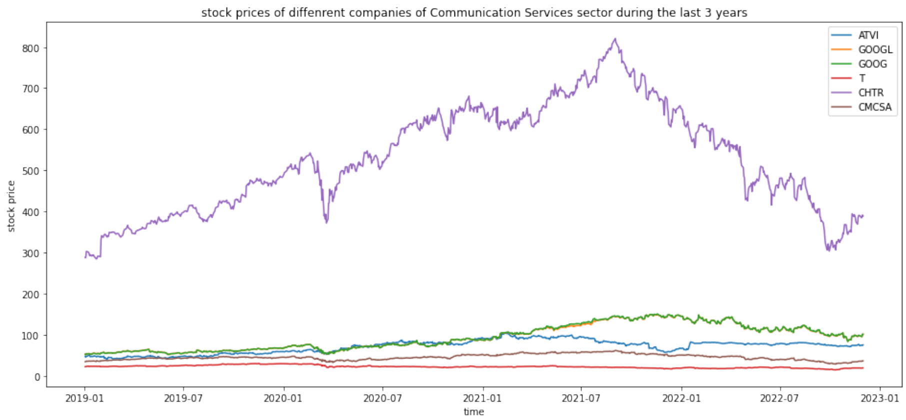


Figure 2.3.4: The trend of stock prices in the Communication Service sector.

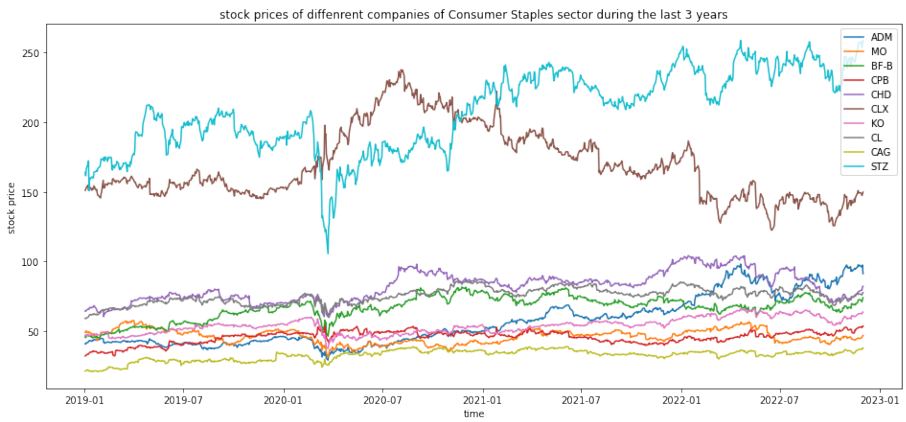


Figure 2.3.5: The trend of stock prices in the Consumer Staples sector.

Chapter 3

Model Establishment

For maximizing and minimizing the profit, we establish two models. The first one is spilt into 2 parts: Stock selection - trading strategy, where we use the Q reinforcement learning to trade off. The second one is split into 3 parts: Prediction with LSTM - Stock selection - trading strategy, where we use the MACD method to trade off.

MODEL 1: STOCK SELECTION & Q LEARNING

3.1 STOCK SELECTION

3.1.1 Assumptions

- Assuming that the log-return of each stock follows normal distribution.
- Assuming that the short selling is not allowed.
- Assuming that there is no risk-free asset.

3.1.2 Methodology

For this part, we use the nonlinear programming method to implement the Markowitz's Mean-Variance Portfolio Opimization (MVO), which can find the optimal weights for each stock. And we preferred maximizing Sharpe ratio to minimizing variance. Sharpe ratio is defined as the difference between the returns of the investment and the risk-free return, divided by the standard deviation of the investment returns. It represents the additional amount of return that an investor receives per unit of increase in risk.
Compared with the Mean-Variance optimation, it is less risk-averse. Since we would train some prediction models and use other trading strategy instead of simply selecting stocks and then randomly buying them at any time, which meant we could control risk to some extent, maximizing Sharpe ratio is a more reasonable.

3.1.3 Implement details

Firstly, to check the first assumption, the kernel density estimate plot of log-return of 50 companies is shown as follows.
As we can see, log-returns of these 50 companies basically follow the normal distribution.
Secondly, We performed a Monte Carlo simulation to generated a large number of portfolio weight vectors. For each simulated allocation, we derive the recorded expected portfolio return and variance. The figure 3.1.2 shows the outcome of the Monte Carlo simulation.
Next, nonlinear programming is performed.

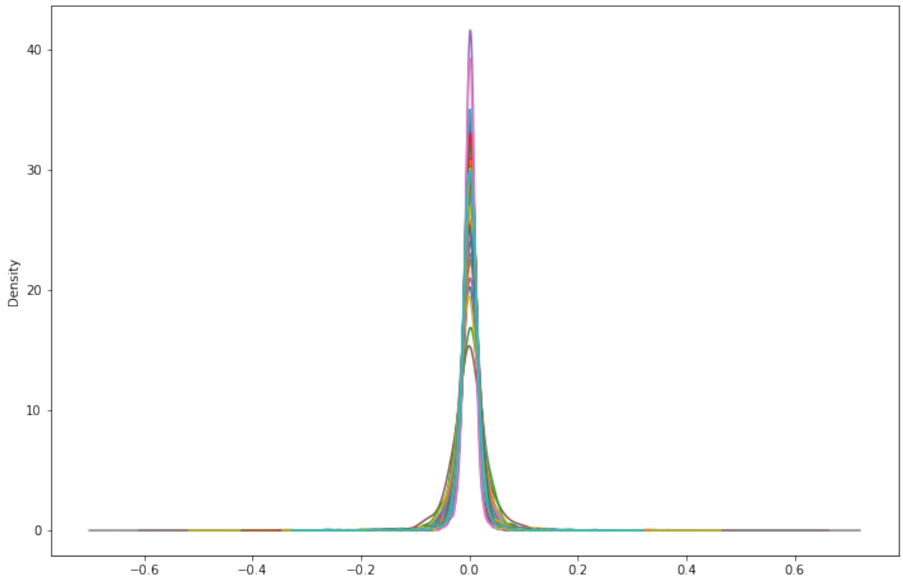


Figure 3.1.1: The kdeplot of log-return.

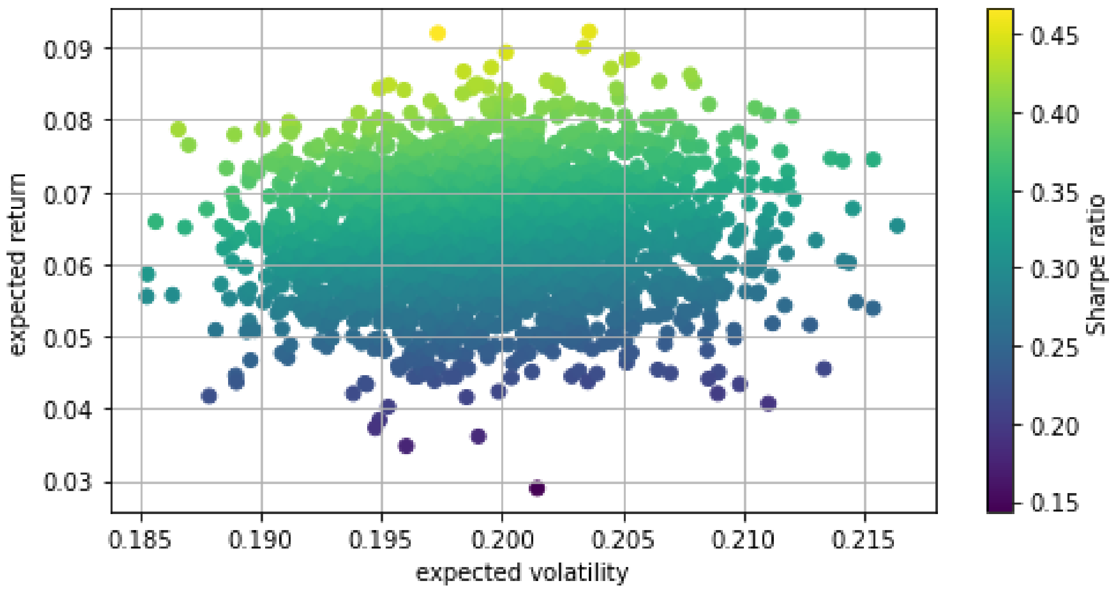


Figure 3.1.2: The plot of expected return against volatility of the portfolio under different weights.

We set the constraint functions are:

$$\sum_{i=1}^{50} w_i = 1 \qquad 0 < w_i < 1, i = 1, 2, ..., 50 \tag{3.1.1}$$

where w_i is the weight of different companies.
Then the return of portfolio is :

$$\mu_p = \sum_{i=1}^{50} w_i * \mu_i \tag{3.1.2}$$

The volatility of portfolio is:

$$\sigma_p^2 = \mathbf{w}^T \Sigma \mathbf{w} \tag{3.1.3}$$

Set the objective function is:

$$SR = \frac{\mu_p - r_f}{\sigma_p} \tag{3.1.4}$$

If we want to get the maximal profit, then we need maximize the objective function, i.e., Sharp ratio. On the contrast, if we want to get the minimal profit, we need minimize the objective function.

3.1.4 Experiment

To implementing the above methods, we use the ***minimize*** function from the ***scipy.optimize*** in Python.
Use log-return of the historical prices five years ago (except the recent three months), substitute into formula (2.1.2), we can derive the average annual log-return. Based on the methods above, we can derive the corresponding weights.
After retaining three decimal places, we select four companies for these two situations, respectively. The results are as follow.
For maximal profit:

Symbol	Company	The weight of company
ADM	Archer Daniels Midland	0.115
AAPL	Apple	0.519
CHD	Church & Dwight Co.	0.214
CTAS	Cintas	0.152

Table 3.1.1: The weights of companies under maximal profit

For minimal profit:

Symbol	Company	The weight of company
MMM	3M	0.261
AAL	American Airlines	0.235
T	AT&T Inc.	0.409
BIIB	Biogen Inc.	0.095

Table 3.1.2: The weights of companies under minimal profit

3.2 Q Reinforcement Learning Strategy

3.2.1 Methodology

The stock trading process can be treated as a Markov Decision Process. Then we transform our goal to a maximization problem. The details are as follows:
State $s = (p, n, b)$, which include the stock prices $p \in R_+^N$, the number of holdings of stocks $h \in Z_+^N$, and the remaining budget $b \in R_+$, where N is the number of stocks that we consider in the trading process and Z_+ denotes non-negative integer numbers.
Action $a \in \{buy, sell, hold\}$: The available actions buy, sell and hold result in increase/decrease, decrease/increase, and no change of the holding of stocks h/remaining budget b, respectively.
Reward $r(s, a, s^{'})$: change of the portfolio value when action a is taken from state s to a new state s^*. The portfolio value is the sum of the budget b and value of all stocks in hold $p^T n$.
Policy $\pi(s)$: the trading strategy of stocks at state s.
Action-value function $Q_\pi(s, a)$: the expected reward achieved by action a at state s by policy π. Our goal is to find a trading strategy to maximize the profit $\Sigma_{t=1}^{t^f-1} r(s_t, a_t, s_{t+1})$ at time t^f. Due to the Markov theory, it can be transformed to find a policy that maximizes the Action-value function $Q_\pi(s_t, a_t)$.

$$Q_\pi(s_t, a_t) = \mathbb{E}_{s_{t+1}}\left[r(s_t, a_t, s_{t+1}) + \gamma \mathbb{E}_{a_{t+1}\sim\pi(s_{t+1})}\left[Q_\pi(s_{t+1}, a_{t+1})\right]\right] \tag{3.2.1}$$

The Q Reinforcement Learning Strategy uses greedy action a_{t+1} which maximizes $Q(s_{t+1}, a_{t+1})$ at state s_{t+1}, which is called the Bellman Equation.

$$Q_\pi(s_t, a_t) = \mathbb{E}_{s_{t+1}}\left[r(s_t, a_t, s_{t+1}) + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1})\right] \tag{3.2.2}$$

The actions are random to begin with but as the steps increase the actions that will yield the highest Q value for that state. The action Q values are calculated through the neural network.

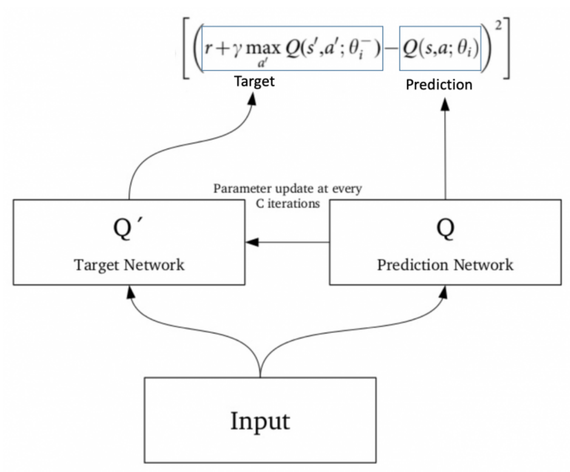


Figure 3.2.1: The flow diagram of Q learning

3.2.2 Implementation details

a. Design of the QLearningDecisionPolicy class
The QLearingDecisionPolicy class maps the inputs to actions using a neural network. It also updates the Q values. Here are two important functions inside the class (these are designed to

maximize profit, for minimizing profit, we just have to change 'max' below to 'min').

Algorithm 1: select action

Initialize ϵ
With probability ϵ select a random action a
otherwise select $a = \max_a Q(s, a)$

Algorithm 2: update q

Initialize reward r according to action a
update Q values and state s:
$s \leftarrow s^{'}$
$Q(s, a) \leftarrow r + \gamma argmax_a(Q(s^{'}, a))$

b. Design of the Simulation Function

The simulation function obtains the current state and calls the select actions function from DecisionPolicy function. This returns the action "Buy", "Sell" or "Hold". Based on the action the budget and portfolio is updated. The next state is found and the policy is updated using the q function. This is repeated for the length of the data minus the history minus 1. The final portfolio, policy and actions of the simulation is then returned.

Algorithm 3: Simulation Function

Initialize the budget, number of holdings of stocks and share value
for *i from 0 to the length of the data minus the history minus 1* **do**
    Obtain current state and portfolio
    call the select action function from QLearingDecisionPolicy class to obtain the action
    if *take Buy action and budget is not less than share value* **then**
        budget ← budget minus share values
        add one to number of holdings of stocks;
    end if
    if *take Buy action and number of holdings of stocks is not less than 0* **then**
        budget ← budget add share values
        minus one to number of holdings of stocks;
    end if
    else
        action ← Hold;
    end if
    update portfolio, reward, state and Q values:
    new portfolio← budget add number of holdings of stocks multiply share value
    reward ← new portfolio minus current portfolio
    state ← combine of new budget, stock prices and number of holdings of stocks
    call the update q function from QLearingDecisionPolicy class to update Q values
end for

3.2.3 Experiments

For each selected company, we train a best policy based on the historical close prices five years ago(except the recent three months, i.e 2017-10-01 to 2022-08-26), and use this policy as our initial policy for our test on close prices of the recent 3 months(2022-8-27 to 2022-11-27). With an initial capital fund of 10000, we first assign the fund for each company, then we compare our reward with baseline reward for each company.

a. Train-Test Split

We Split the data of close prices of selected companies from 2017-10-01 to 2022-11-27 to Train-set (2017-10-01 to 2022-08-26) and Test-set (2022-8-27 to 2022-11-27).

b. Train

For each company, by using the simulation function, we simulated 20 times to find the best policy (which maximize the portfolio value) on the Train-set.

c. Test

For each company, with an assigned fund, we use its best policy on the Train-set to test the performance on the Test-set. The results for maximal profit are :

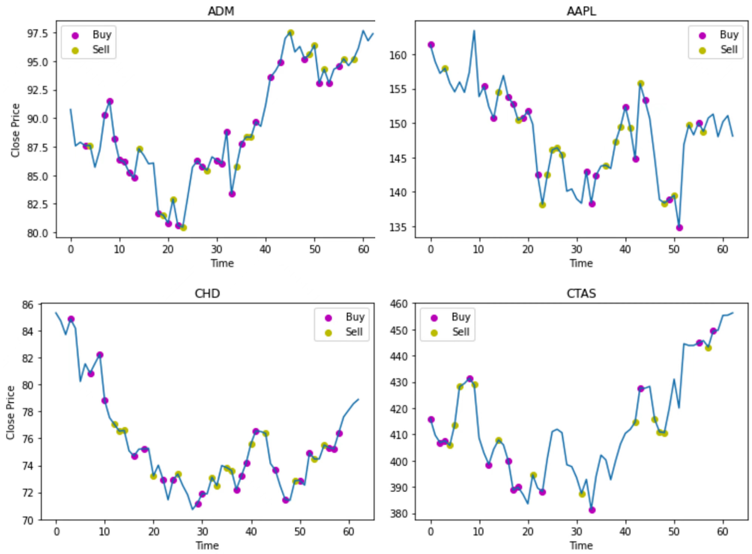


Figure 3.2.2: Q learning for maximizing the profit

The results for minimal profit are:

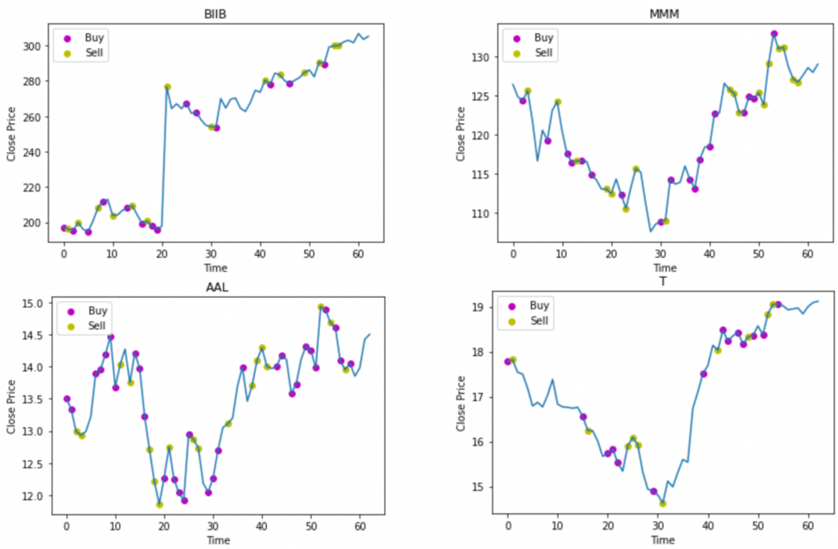


Figure 3.2.3: Q learning for minimizing the profit

For our Q-learning strategy, we don't do any prediction in the beginning. Finally we can earn up to around $300 profit and lose up to $41, both of which are quite small. And from the plot we can see that the polices which were well trained on our Train-set didn't performs well on the

Test-set. Sometimes it makes some bad actions such as buying stocks near the peak of stock price.

As Model 1 does not perform well, we decide to introduce the second model, which we will give some detailed introduction as part 2 shows.

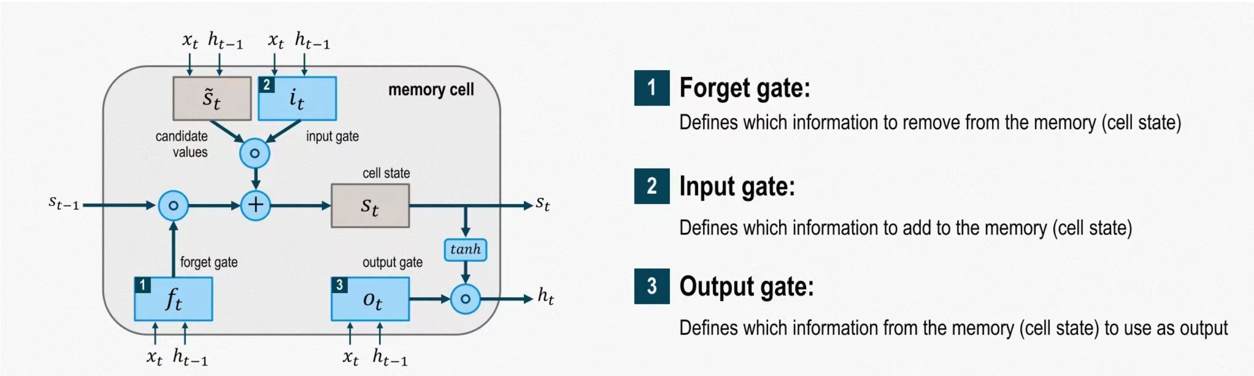
MODEL 2: LSTM PREDICTION & MACD STRATEGY

In this model, we first predict the last 3 month stock prices for 50 companies using the historical 5 years stock prices, and then we use Markowitz method to select optimal stocks into our portfolio. Finally we use MACD method to finish our trades-off.

3.3 LSTM FOR STOCK PRICE PREDICTION

3.3.1 Methodology

Long short term memory (LSTM) networks is one type of recurrent neural network which is used to learn order dependence in sequence prediction problems. LSTM are capable of overcoming the previously inherent problems of RNNs such as vanishing and exploding gradients. LSTM networks are comprised of an input layer, several hidden layers, and an output layer. The most important characteristic of LSTMs is memory cells contained in the hidden layers. The figure below illustrates the structure of an LSTM memory cell. As we can see, for each memory cell, x_t and h_t correspond to the input and hid-den state respectively, at time t , and i_t , o_t and f_t, are the gates which are called input, output and forget gates, respectively, s_t is adjusting its cell state. It is worth noting that the input gate decides which data can be added into the memory cell, the output gate decides which data from the memory cell can be used as output, and the forget gate decides which data should be deleted from the memory cell.



Structure of LSTM memory cell following Graves (2013) and Olah (2015).

Figure 3.3.1: Structure of LSTM memory

The calculations for each state and gate are performed as the following formulas:

$$
\begin{aligned}
&f_t = \text{sigmoid}\left(W_{f,x}x_t + W_{f,h}h_{t-1} + b_f\right)\\
&\tilde{s}_t = \text{sigmoid}\left(W_{\tilde{s},x}x_t + W_{\tilde{s},h}h_{t-1} + b_{\tilde{s}}\right)\\
&i_t = \text{sigmoid}\left(W_{i,x}x_t + W_{i,h}h_{t-1} + b_i\right)\\
&s_t = f_t * s_{t-1} + i_t * \tilde{s}_t\\
&o_t = \text{sigmoid}\left(W_{0,x}x_t + W_{o,h}h_{t-1} + b_o\right)\\
&h_t = o_t * \tanh\left(s_t\right)
\end{aligned}
$$

x_t is the input vector at timestep t. $W_{f,x}$, $W_{f,h}$, $W_{\tilde{s},x}$, $W_{\tilde{s},h}$, $W_{i,x}$, $W_{i,h}$, $W_{o,x}$ and $W_{o,h}$ are weight matrices. b_f, $b_{\tilde{s}}$, b_i and b_o are bias vectors. f_t, i_t and o_t are vectors for the activation values of the respective gates. s_t and $\tilde{s}_t$ are vectors for the cell states and candidate values. h_t is a vector for the output of the LSTM layer.

3.3.2 Implementation details

The Implementation details are divided into 3 parts: data preparation, model establishing, model fitting and evaluation.

a. data preparation

First, we split the data into Train-set (past 5 years except last 3 months) and Test-set (last 3 months). Second, we store the data in the form of a time window. Then, for data in each window, we conduct a min-max normalization. The above steps are done to both Train-set and Test-set respectively.

b. model establishing

Our LSTM model is shown as follows:

Layer(type)	Output Shape	Number of Parameters
LSTM	(None,9,50)	10400
Droupout	(None,9,50)	0
LSTM	(None,9,50)	20200
LSTM	(None,50)	20200
Droupout	(None,50)	0
Dense	(None,1)	51

It is worth noting that in order to use more complete information to train a better model, we use all the data in the training set data (including open, high, low, close, adjusted close and volume) to model to estimate the close price on the test set.

c. model fitting and evaluation

Adam is used as the optimiser to improve the neural network. After random search in optimising the parameters: (1) the sequence length, ranging from 5 to 20; (2) the number of epochs, ranging from 100 to 600; (3) neuron activation function; (4) the number of neurons per hidden layer, ranging from 2 to 200. We find our best parameters: (1) the sequence length 10; (2) the number of epochs 500; (3) neuron activation function linear; (4) the number of neurons per hidden layer 50. MSE is used as our model evaluation criteria.

3.3.3 Experiments

For the purpose of stock selection for investment in the future, we predicted the close prices for all 50 companies. That is, the stock price corresponding to each ticker is fed into the model one at a time, and predictions are made for this stock using the parameters trained on the same. Here are some of our results.

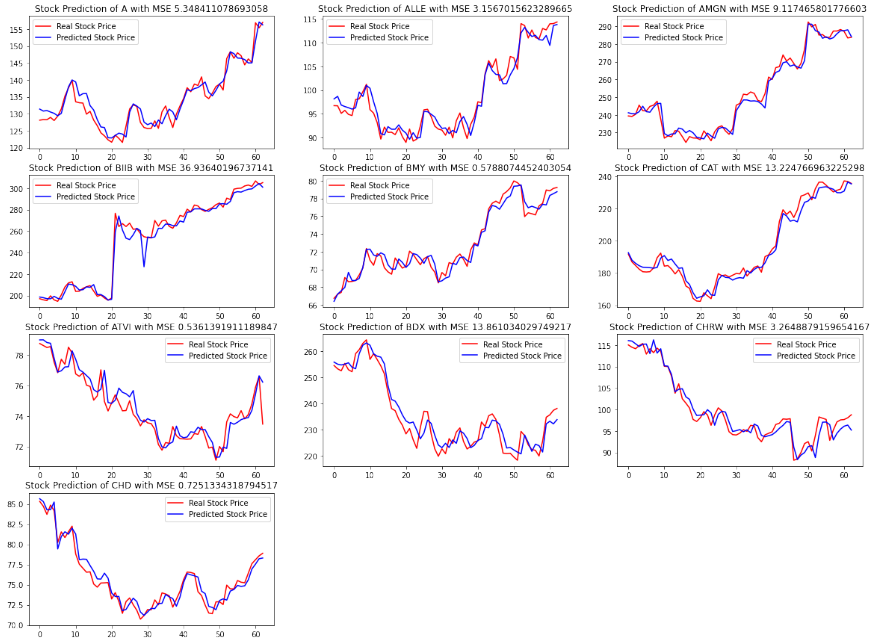


Figure 3.3.2: The fitted plots of LSTM

We can see from the plot and MSE that the prediction of stock price by LSTM is accurate to some extent. However, we can also find that there exist a certain hysteresis in predicting the real data by using LSTM. This maybe due to the sequence autocorrelation, which we will discuss in the conclusion part.

3.4 MACD Strategy

3.4.1 Methodology

In this project, the technical indicator we use is MACD, Moving Average Convergence Divergence. It is a trend-following leading indicator that is calculated by subtracting two Exponential Moving Averages (one with longer and the other shorter periods). There are three notable components in a MACD indicator.

(1) MACD Line : This line is the difference between two given Exponential Moving Averages. To calculate the MACD line, one EMA with a longer period known as slow length and another EMA with a shorter period known as fast length is calculated. The most popular length of the fast and slow is 12, 26 respectively. The final MACD line values can be arrived at by subtracting the slow length EMA from the fast length EMA. The formula to calculate the MACD line can be represented as follows: MACD LINE = FAST LENGTH EMA - SLOW LENGTH EMA.

(2) Signal Line : This line is the Exponential Moving Average of the MACD line itself for a given period of time. The most popular period to calculate the Signal line is 9. As we are averaging out the MACD line itself, the Signal line will be smoother than the MACD line.

(3) Histogram : As the name suggests, it is a histogram purposely plotted to reveal the difference between the MACD line and the Signal line. It is a great component to be used to identify trends. The formula to calculate the Histogram can be represented as follows: HISTOGRAM = MACD LINE - SIGNAL LINE.

In this project, we are going to build a simple crossover strategy that will reveal a buy signal whenever the MACD line crosses above the Signal line. Likewise, the strategy will reveal a sell signal whenever the Signal line crosses above the MACD line.

3.4.2 Implement Details

MACD Calculation

In this step, we are going to calculate all the components of the MACD indicator from the extracted historical data of the selected stocks with best performance which we have calculated before.
Firstly, we define a function that takes the stock's price, length of the slow EMA, length of the fast EMA, and the period of the Signal line. Inside the function, we are first calculating the fast and slow length EMAs. Next, we calculated the values of the MACD line by subtracting the slow length EMA from the fast length EMA and stored it into the MACD variable. Followed by that, we defined a variable to store the values of the Signal line calculated by taking the EMA of the MACD line's values for a specified number of periods. Then, we calculated the Histogram values by subtracting the MACD line's values from the Signal line's values and stored them into the HIST variable.
Using the created function, we stored all the MACD components that are calculated from the stock price of each stock.

MACD Plot

First, we plot all the MACD components of selected stocks (here we take ADM as an example, but it is not included in our portfolio).

Figure 3.4.1: MACD component of ADM

We can notice that the plot of the calculated Histogram values turns red whenever the market shows a negative trend and turns green whenever the market reveals a positive trend. The Histogram plot spreads larger whenever the difference between the MACD line and the Signal line is huge, and it is noticeable that the Histogram plot shrinks at times representing the difference between the two of the other components is comparatively smaller. The next two components are the MACD line and the Signal line. The MACD line is the grey-colored line plot that shows the difference between the slow length EMA and the fast length EMA of stock prices. Similarly, the blue-colored line plot is the Signal line that represents the EMA of the MACD line itself. Like we discussed before, the Signal line seems to be more of a smooth-cut version of the MACD line because it is calculated by averaging out the values of the MACD line itself.

3.4.3 Experiments

Create the Trading Strategy

In this step, we are going to implement the discussed MACD trading strategy. First, we define a function which takes the stock prices and MACD data as parameters. After that, we implement the trading strategy through a for-loop. Inside the loop, If the condition to buy the stock gets satisfied, the signal value will be appended as 1 representing to buy the stock. Similarly, if the condition to sell the stock gets satisfied, the signal value will be appended as -1 representing to sell the stock.

Plot the Trading Lists

The created trading lists are plotted using arrows in the following graph.

Figure 3.4.2: MACD signals of ADM

We plot the MACD components along with the buy and sell signals generated by the trading strategy. We can observe that whenever the MACD line crosses above the Signal line, a buy signal is plotted in green color, similarly, whenever the Signal line crosses above the MACD line, a sell signal is plotted in red color.

Create Position

Next, we will implement the strategy on the last three months of data obtained from the LSTM prediction. We create a list that indicates 1 if we hold the stock or 0 if we don't own or hold the stock. We take the stock ADM as an example here (see as table 3.4.1 shows).
From the output being shown, we can see that in the first row our position in the stock has remained 1 (since there isn't any change in the MACD signal) but our position suddenly turned to 0 as we sold the stock when the MACD trading signal represents a sell signal (-1).
We implement our MACD trading strategy on predicted prices over all 6 selected stocks. (The selection method is the same as the **Section 3.1**, the only difference is that we use the predicted stock prices of last 3 month to caculate the average return, so we will not go into details here.)
The outcome is shown below:

Date	Close	macd	signal	macd_signal	macd_position
2022/8/29	90.47	0.00	0.00	0	1
2022/8/30	90.47	-0.25	-0.05	-1	0
2022/8/31	88.68	-0.43	-0.13	0	0
2022/9/1	88.69	-0.58	-0.22	0	0
2022/9/2	88.69	-0.69	-0.31	0	0
2022/9/6	87.25	-0.92	-0.43	0	0
2022/9/7	86.38	-0.97	-0.54	0	0
2022/9/8	86.87	-0.75	-0.58	0	0
2022/9/9	90.14	-0.48	-0.56	1	1
...	...	...	...	...	...
2022/11/7	95.27	2.67	2.23	0	1
2022/11/8	96.15	2.67	2.31	0	1
2022/11/9	95.77	2.37	2.32	0	1
2022/11/10	93.31	2.20	2.30	-1	0

Table 3.4.1: Positions of ADM

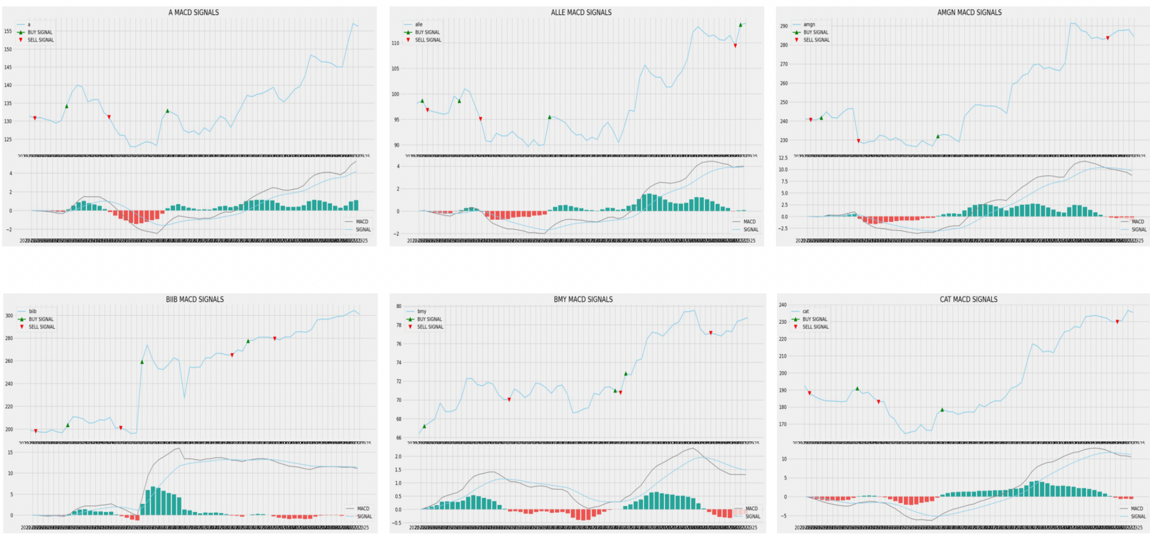


Figure 3.4.3: MACD trading for maximizing the profit

Stocks in Portfolio	Weight in portfolio	Profit of MACD strategy
A	32.67%	14%
ALLE	11.88%	9%
AMGN	29.70%	15%
BIIB	6.93%	2%
BMY	11.88%	11%
CAT	6.93%	18%

Table 3.4.2: MACD Performance

We test our strategy by investing ten thousand USD into our trading strategy. We calculate the number of stocks of each asset we can buy using the investment amount. Then, we find the investment returns followed by some data manipulations tasks.
As can be seen from the results, the returns did not perform well due to the recent market shocks. However, below we will compare the performance with that of ETF and we can see that the portfolio we have constructed has outperformed the broader market.
If we select the bad-performed stocks to build the portfolio (still using Markowitz method to minimize Sharpe ratio), and take the opposite approach to the previous one, i.e. we buy when signal is -1 and sell when signal is 1, we end up with the following result:

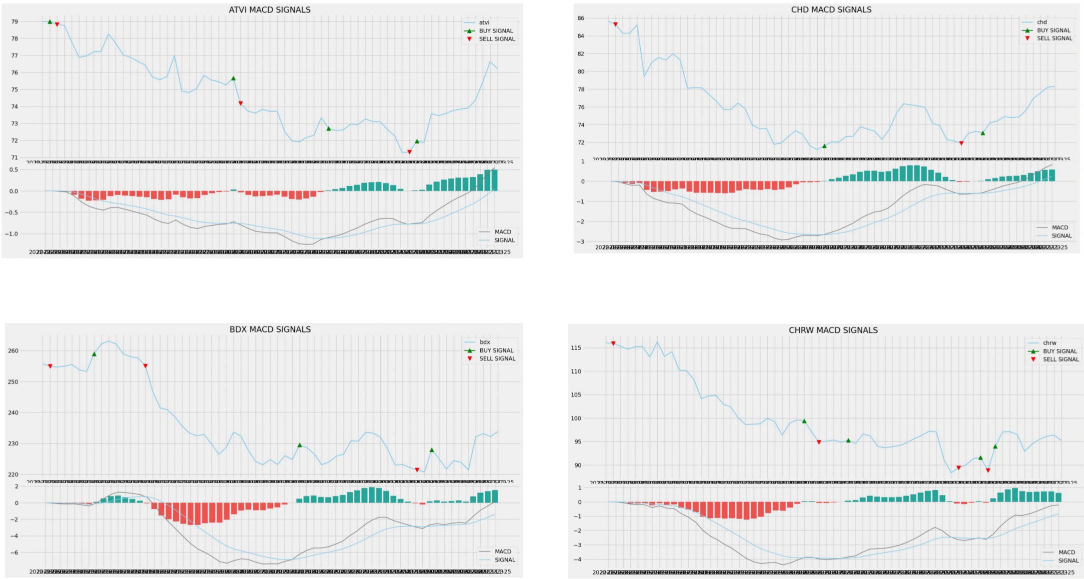


Figure 3.4.4: MACD trading for minimizing the profit

Stocks in Portfolio	Weight in portfolio	Minimum Profit
ATVI	10.1%	-5%
BDX	8.2%	-6%
CHRW	59.1%	-8%
CHD	22.6%	-15%

Table 3.4.3: Worst Performance

As you can see, the final return is significantly reduced. By reversing the operation, the final return on investment obtained was -8.73%, which means we lost $873 when investing $10000. Thus our strategy can be considered effective.

SPY ETF Comparison

In this part, we can get an idea of how well our trading strategy performs against a benchmark (SPY ETF). We compare the returns we get from the SPY ETF with our MACD strategy returns on each stock (assume the initial investments amount are the same between SPY ETF and the selected portfolio). After executing the MACD strategy on the predicted 3-month data set, we obtain the following results:
This step's procedure is similar to the one used in the previous back testing step. But instead of investing in a certain stock, we invest in SPY ETF by not implementing any trading strategies.

Benchmark Profit	$102.72
Benchmark Profit percentage	1%
Maximal Portfolio Profit	$2347.28
Maximal Portfolio Profit percentage	23.47%
Minimal Portfolio Profit	-$873.58
Minimal Portfolio Profit percentage	-8.73%

Table 3.4.4: Portfolio Performance Compared with Benchmark

From the output, we can see that our MACD trading strategy has outperformed the SPY ETF by 22.47%. And our model 2 has outperformed model 1 more than 20%, so it is better to choose model 2 as make money strategy.

3.5 Conclusion

This part (for Q4) mainly establishes models on how to make money in stocks for short-term investing, and we establish two models. Both models can be roughly divided into two parts: selection of stocks, trading strategy. The stock selection methods in both 2 models are basically same. However, the difference is that whether we use the prediction information.

In the first model, we establish the model without prediction. We just simply select our portfolio based on the historical return information, which means we discard a lot of information such as the stocks' trend and distribution. And thus, we just get 2%-4% return for maximizing profit, and -0.4% for minimizing profit. As the problem become much more complicated when considering all combinations of stocks at each time for one agent (In fact we have tried this even using more advanced algorithms such as A2C, DDPG and PPO but got bad performance), we limited the state space by only considering one stock for one agent. In this way, for each agent at each time, the state space is {Sell, Buy, Hold}, which is much more simple and practical (also with better performance).
We do find that Q-learning performs better than the baseline strategy (which is to use all capital fund for the company to buy stocks at first, then sell them all out in the end). However, after optimization, the Q-learning still performs bad in making money. Here are some of the reasons and limitations of our work now:
a. First, our fund are assigned in the beginning and kept unchanged to the end for each agent. However, considering the changes of stock market in our time range, it is more reasonable to design a mechanism (such as some indicators) to reallocate funds for each stock(agent).
b. Second, it is not enough to only consider historical data for making stock trading strategy, real-time news and important events are essential in making strategy. So our future work is to use some text mining techniques on social media to provide more information which can help us to predict the trend of the stock market and formulate the optimal strategy.
c. Lastly, some data cleaning and processing might be helpful before training our model. We have studied the reasons why these policies didn't perform well, and found that an very important reason is that compared to historical data, data in last 3 months of these selected companies Shows different trends and distributions. As when it comes to formulating future strategies, the contribution of data at different times to the model is different. For example, the importance of recent data might be bigger than old data. So one another future work is to add a mechanism to consider the importance of data in our Q learning algorithm.

IN THE SECOND MODEL , as we can see, model fits well, and we use much more information (all information from stocks, including open, high, low, close, adjusted close and volume) compared with model 1. And based on the predicted price to select stocks, after MACD trading strategy, the maximal return increases from 2% to 23.47%, which means model 2 is very effective. By contrast, the minimal return is -8.73%, which means we lose $ 873.58

However, there's still some problems we have to solve.

Firstly, for LSTM part:

a. The general model doesn't consider the correlations of stocks from the same sector. In fact, there do exist correlations within sectors of stock prices, which are shown as follows.

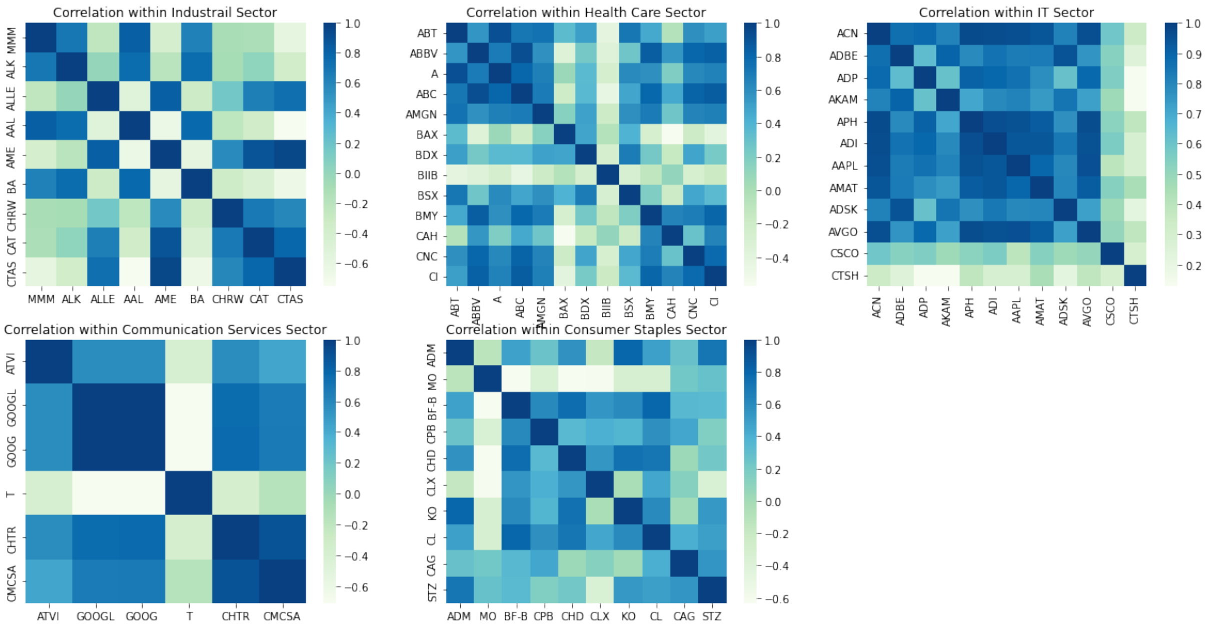


Figure 3.5.1: The correlation plot in each 5 sectors

Because of this, a model considering correlation within sectors needs to be trained specifically to each sector, by regarding all historic data of all stocks within that sector as the Train-set. Specifically, we need to establish a sector-considered model which is just a simple superimposition of LSTM models but with Train-set prepared in a different way.

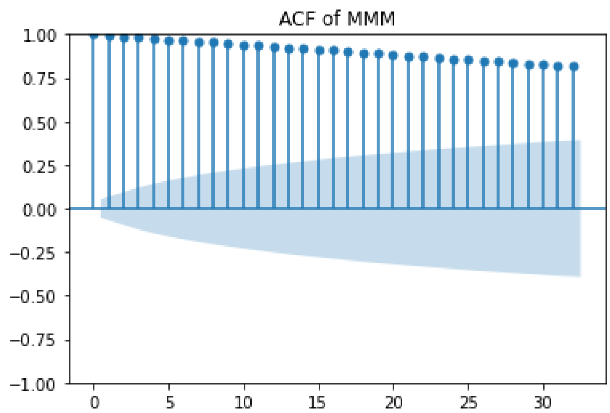


Figure 3.5.2: The ACF plot of MMM

b. Although the prediction of stock price by LSTM is accurate to some extent, it is not difficult to find that LSTM has a certain hysteresis in predicting the real data, which may be due to the autocorrelation of sequence samples. One example is as Figure 3.5.2 shows:
This inspires us to combine the LSTM model with ARIMA model which considers sequence autocorrelation. And this is what we can try in the future.
For MACD part, in volatile market conditions, the MACD strategy has a certain lag in judging the market and is not effective in hedging risk. Therefore, it is necessary to choose a more appropriate smoothing period based on the risk aversion of the investor, combined with a more elaborate algorithm. Moreover, MACD has the tendency to reveal false trading signals. So, we can use a technical indicator in addition to MACD to cross verify whether the represented signal actually is an authentic trading signal.
c. Considering the hysteresis of LSTM in predicting the real stock price, we revised the StockBot decision making algorithm in Bibliography 2. Specifically, At time t, we make decisions based on the curvature at time $t-1$ rather than time t. And we also Optimized the number of shares per transaction. Compared with the initial algorithm, the profit changes from $1900 to about $4900 with an initial fund of 10000. Since our team only got this result a few hours before due, it was too late to write the report, so the algorithm is briefly described as follows:

Algorithm 4: Revised StockBot decision making algorithm

1 Obtain the predicted trajectory, c (stock forecast for forward look days)
2 Compute the nature of change: $\delta_i = \text{sign}\,(c_i - c_{i-1})$
3 Compute the curvature of the forecast: $\Delta_i = \delta_i - \delta_{i-1}$
4 Make the decision: $\text{Decision} = \begin{cases} \Delta_i = 2 \rightarrow \text{ sell (Indicates local maxima)} \\ \Delta_i = 0 \rightarrow \text{ hold (Indicates no change)} \\ \Delta_i = -2 \rightarrow \text{ buy (Indicates local minima)} \end{cases}$

Chapter 4

Impact on Stock Market

4.1 METHODOLOGY

For the Q5, We first find the lowest point of the V-shaped rebound trend as the turning point of the stock price based on the stock price trend of each company in the five industries obtained in Section 2.2. Then we determined the exact time when the turning point occurred and crawl the financial news that occurred during this time interval in the Wall Street Journal as quantitative evidence. Finally, to find out the events that might affect the stock price when the turning points occurred, after pre-processing the crawled news headlines, the most frequently occurring words were counted, and in order to more intuitively see the keywords with the highest frequency in all news titles, we cited word clouds for quantitative analysis.

4.2 IMPLEMENTATION DETAILS

Firstly, based on the plots of the trend of stock prices of different companies in five sectors during the last three years in section 3, it can be clearly seen that the stock price trends of different companies in the different sectors have different degrees of V-shaped rebounds. Therefore, the rebound point(i.e. the lowest point approximately), was used as a turning point for subsequent analysis.
For convenience, we found the stocks with the most pronounced V-shaped trend ("STZ", "AVGO", "CI", and "BA" respectively) in the industrial sector, Health Care sector, IT sector and Consumer Staples sector (Communication Services sector is not considered because it is more difficult to find the turning points of the stock prices of individual companies). And the lowest point of the four stocks' prices were output and the time obtained as the turning point, the results were as follows:

Stock	Time
STZ	2020-03-23
AVGO	2020-03-28
CI	2020-03-23
BA	2020-03-20

Table 4.2.1: Time when turning points occur

Secondly, based on the stock price turning points of different sectors obtained in tabel 4.2.1, it can be clearly seen that the points concentrated during the period from March 18, 2020 to March 23, 2020. Since the opening price of a stock is related to the previous day's economic activity, at the same time, the next day's financial news also reflects the economic events that

affected the previous day's stock price volatility. Therefore, in order to analyze more concretely the events that affect the turning point of the stock price, we expanded the time interval.
"The Wall Street Journal" is a comprehensive newspaper featuring financial reports, focusing on financial and commercial reports, and has a wide range of influence in the world, and its content is enough to affect daily international economic activities. Based on the above advantages and in order to automatically extract events that might affect the stock price on social media when the turning points occur, the Wall Street Journal's global economic news from March 17, 2020 to March 24, 2020 were crawled.
In addition to crawling news titles, to strengthen the initial understanding of the news and find the original news content more conveniently, it was also necessary to crawl news links and news profiles. In the process of solving this problem, we also realized the above functions.
Then, in order to be able to comprehensively observe and extract the main events that affect the stock price during this period, it was very helpful to combine the financial news of each day into one table for subsequent further analysis. At the same time, the merged news will also be used as quantitative evidence for the next step of quantitative analysis.
Through a general observation of the financial news of these eight days, it was obvious that the term coronavirus appeared very frequently, so it can be preliminary inferred that when the inflection point occurred, the main event that affects the stock price was the outbreak of new coronavirus pneumonia, in order to verify this inference, the next step of quantitative analysis was needed.
Next, based on the quantitative evidence and preliminary inference obtained in above description, we extracted news headlines from the merged table, and then converted the extracted headlines into text format.
Finally, in order to verify the preliminary inference that when the inflection point appeared, the main event that affected the stock price was the outbreak of new coronavirus pneumonia, we segmented the news titles in the processed text document and counted the word frequency. And then we select the top 5 words with the highest frequency for output, and the output result was as follows:

Word	Frequency
Coronavirus	206
New	34
York	18
City	12
Stimulus	12

Table 4.2.2: The plot of top 5 high frequency words

In addition to prepositions, it was obvious that the word with the highest frequency was coronavirus. In order to more intuitively see the keywords with the highest frequency in all news titles, a word cloud map was made.

4.3 Experiments

The word cloud map was as follows:
From the above word cloud map, we can see intuitively that coronavirus is the word with the highest frequency, which also confirms the previous inference. therefore, when the inflection point occurs, the main event that affects the stock price is the outbreak of new coronavirus pneumonia.
The impact of coronavirus on the sectors:

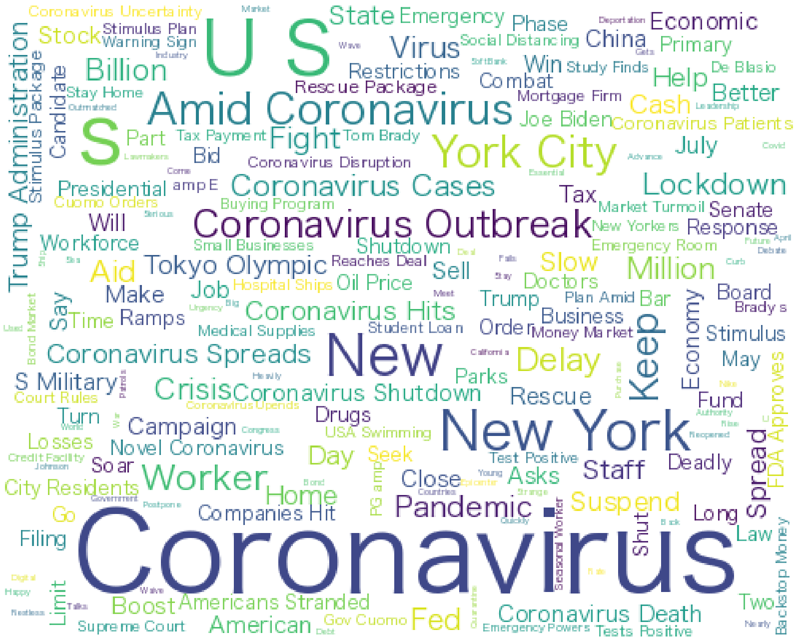


Figure 4.3.1: Word cloud map of events.

- Under the influence of the epidemic, the manufacturing industry is in trouble due to labor difficulties, rising costs, and reduced orders.
- The impact of the epidemic on the global supply chain is especially manifested in the electronics field. In the field of electronic products, top semiconductor and electronics companies such as Samsung, Samsung SDI, and LG Electronics are concentrated in Market, the hardest-hit area of the epidemic in South Korea; Japan accounts for about 50 percentage of the world's production capacity of silicon wafers, the core material of semiconductors. Affected by the epidemic, major production in Japan and South Korea has come to a standstill , directly affect the global industrial supply chain.
- At the same time, there will also be industries that are positively affected by the epidemic. Because of the increasing demand for anti-cold and pneumonia treatment drugs, disinfectant and alcohol, the pharmaceutical industry will experience short-term stimulating development.

The impact of coronavirus on the whole stock market:
The large-scale outbreak of the epidemic, the sharp drop in crude oil and other markets, and the quantitative easing policies generally adopted by developed countries in response to the epidemic may lead to the accumulation of financial risks and large-scale turmoil; the stock markets of many economies have triggered circuit breakers, and U.S. stocks have been in March. There were four circuit breakers in ten days; the stock markets of many countries plummeted for several consecutive trading days, the largest decline in history, which had a profound impact on the global stock market.

Bibliography

[1] Wang W, Li W, Zhang N, et al. Portfolio formation with preselection using deep learning from long-term financial data[J]. Expert Systems with Applications, 2020, 143: 113042.

[2] Liu X Y, Xiong Z, Zhong S, et al. Practical deep reinforcement learning approach for stock trading[J]. arXiv preprint arXiv:1811.07522, 2018.

[3] Mohanty S, Vijay A, Gopakumar N. StockBot: Using LSTMs to Predict Stock Prices[J]. arXiv preprint arXiv:2207.06605, 2022.