Will Technical Analysis Price Patterns perform well?
—— Experiments based on Kaggle Stock Dataset

Chenhao Fang
cfang45@wisc.edu

Chenyang Jiang
cjiang7748@wisc.edu

Shikun Liu
sliu673@wisc.edu

Haoran Ten
hteng22@wisc.edu

Zixiang Xu
zxu475@wisc.edu

1. Introduction

The Democratic candidate Joe Biden captured enough Electoral College votes to win the US Presidency (subject to recounts and court challenges), with the market now turning its attention to the implications of a divided government. Many investors are trying to gain more profit in this special time. Among many famous investing methods, there is one method called Technical Analysis Price Patterns draws our group's attention.[1] It has been widely used to use price patterns to examine current movements and forecast future market movements. In this method, Patterns are the distinctive formations created by the movements of security prices on a chart and are the foundation of technical analysis. A pattern is identified by a line that connects common price points, such as closing prices or highs or lows, during a specific period of time. These patterns can be as simple as trendlines and as complex as double head-and-shoulders formations. Technical analysts seek to identify patterns as a way to anticipate the future direction of a security's price. The figure 1 shows a pattern consisting of triple bottoms on the daily McGraw Hill chart, leading to a trend reversal[1].

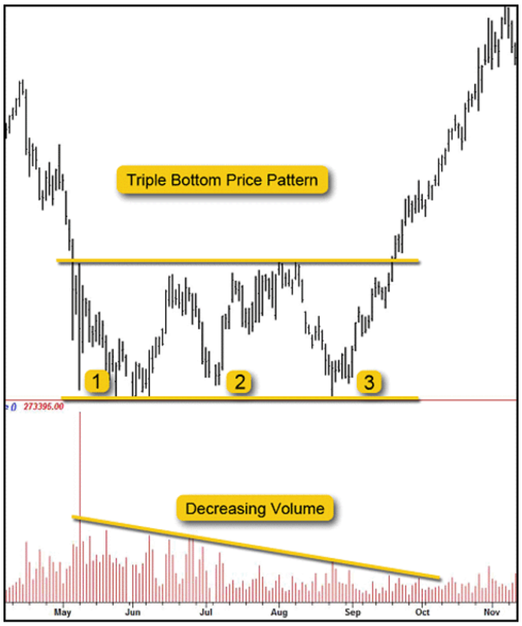

Figure 1. One pattern in the McGraw Hill chart

The problem is, this methods will only see the patterns in the stock charts, which is drawn by using the stock price every day. It ignores all the other factors related to that company. Also, some stock patterns were happened many years ago, and some investors will try to identify that pattern in the stock chart nowadays. Is this method, that identifies a pattern many years ago and only consider stock price while ignoring all other factors, really reasonable? We select this question as our project topic and we run experiment to verify if this method will work.

In this project, we aim at this problem and use a large stock data set from Kaggle[2] to run experiment. Our work consists of effective and comprehensive data cleaning and large scale distributed computing. We first deal with missing values and wrong values in the stock dataset. Then we select one representative stock - Telsa and find the time period before it soars. We use its time series in that time period as our target pattern. Then we tried various methods to measure the similarity between time series. Several measurements are used in this project, including correlation, Euclidean distance, Mahalanobis distance and pattern model representation (PMR). Since the dataset is more than 10GB, we use computing resources from Center for High Throughput Computing at the University of Wisconsin -

[1] Source: Investopedia. Chart created with TradeStation 9.0.

Madison and all our experiments were running in parallel on this platform.

In the following section, we briefly introduce several researchers' work that is related to this problem and provides us insights. In Section 3 we introduce our dataset and the methods and concepts we used in our projects. In section 4, the implementation of experiments is described. Final results, discussions and conclusions are involved in section 5 and 6.

2. Related Work

In this dataset, searching for similarity between time series plays an important role. In [3], the paper compares different methods of measuring similarity in long time series and presents our analysis in terms of accuracy and precision when various forced time series variations are imposed. In another paper [4], the authors proposed a new method that obtains relevant features from financial time-series based on the intrinsic information existing inside stock market time series based on their similarity and employing Metric Access Methods to speedup the process. These works may be a good reference for us to choose the proper measurement.

3. Statistical Methods

3.1. Framework Overview

In this project, our goal is to find out stocks with high similarities with our target stock. Our solution to this competition contains four steps, including identifying the issue, data processing, calculating similarities and evaluation. The similarities measurements utilized are correlation, Euclidean distance, Mahalanobis distance and pattern model representation (PMR).

3.2. Data Description

In this project, the dataset comes from Kaggle, which is a large platform for data science competition. You can access the data through this by ***clicking me***. This dataset contains approximately 100k stocks and their price from 1st Jan 1996 to 7th Aug 2020. The size of this dataset is more than 10GB and therefore meets our requirement for this project.

Understanding the structure of the original data is key to processing data. This dataset contains two parts: One is Tickers.xlsx. This file consists of Tickers' names for more than 1,00,000 companies across the globe. This file contains six columns and Table 1 gives some brief information about the data.

The other is Data folder.This Directory Consist of more than 1,00,000 directory, in which company data is Stored. Each Company has separate directory for its data file.The variables in the Data Folder are as follows:

We decide to choose Close in our project as the measurement of stock price. The figure 2 shows a stock chart that is drawn by using the close price.

Names	Definition
Ticker	Ticker name
Name	Name of company
Exchange	Short name of Exchange
Category	Category of business
Country	The registering country

Table 1. Details of Data Format

Names	Definition
Date	The date of record
Open	The open price in a day
Close	The price at the end of a day
Low	The lowest price in one day
High	The highest price in one day

Table 2. Details of Data Format

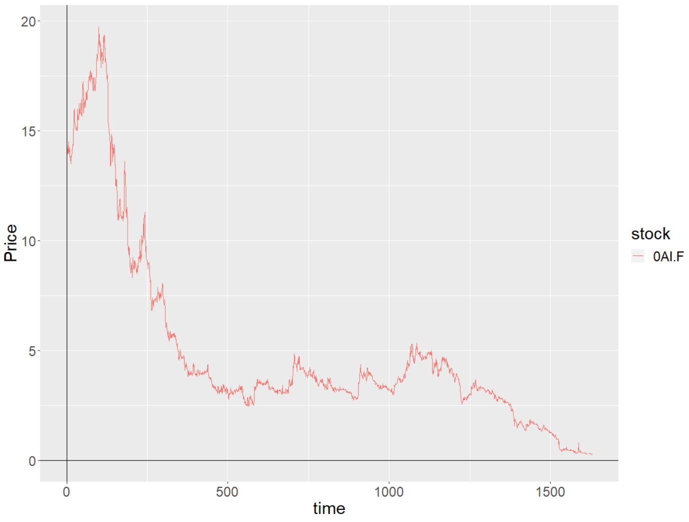

Figure 2. A sample stock chart

3.3. Data Processing

3.3.1 Deal with missing values

We count the number of samples that have missing time series in the whole dataset. It can be observed that the total number of stocks is 100k and the number of stocks that are missing is 134. We write scripts to get the right stock data from Yahoo Finance and replace all the missing data.

3.3.2 Deal with wrong values

Some stocks in the dataset have extremely large price, i.e. 1 million Dollars per share yesterdays and 1 Dollar today. That's impossible. We refer to the Yahoo Finance and replace them with the right price. Luckily, there are only a limited amount of files that contains the wrong value. We replace the wrong value with the real one.

3.3.3 Standardize the data

Since we only want to find stocks that looks similar to our target stock, the absolute value of the price does not matter. Therefore, we standardize the time series before calculating their similarities.

3.3.4 Others

Another thing to note is that there are several folders like AAPL, AAPL34F.SA, AAPL34.SA, AAPL.BA, AAPL.MX, AAPL.SW. They are Apple Inc prices in different exchanges around the globe and are calculated in different currencies. Therefore, when we find stocks with high similarities, we need to remove duplicates like this.

3.4. Similarity Measurements

3.4.1 Pearson Correlation Coefficient

The Pearson Correlation Coefficient is the most famous similarity measure that is invariant to shifting and scaling being expressed by

$$r_{CC}(X(t), Y(t)) = \frac{\sum_{t=1}^{N}(X(t)-\mu_X)(Y(t)-\mu_Y)}{\sqrt{\sum_{t=1}^{N}(X(t)-\mu_X)^2}\sqrt{\sum_{t=1}^{N}(Y(t)-\mu_Y)^2}}$$

In this expression, N is the length of the time series and μ is the length of each time series. The Pearson Correlation Coefficient range is $-1 \leq r \leq 1$ where 1 indicates a perfectly matched between two time-series and a value less than 0 indicates a negative association. 0 indicates that there is no association between the two variables.

3.4.2 Euclidean Distance

The Euclidean Distance is calculated as follows:

$$D_{\text{Euclidean}}(\mathrm{X(t)}, \mathrm{Y(t)}) = \sqrt{\sum_{\mathrm{t}=1}^{\mathrm{N}} |\mathrm{x_t} - \mathrm{y_t}|^2}$$

The Euclidean distance has limitations. It does not allow different sequence's length, different sampling rates, shifting in time axis even though these time series are similar to each other. These drawbacks make the Euclidean distance difficult for direct use.

3.4.3 Mahalanobis distance

The Mahalanobis distance defined a dissimilarity measure between two time-series with the same distribution and covariance matrix S.

$$D_{\text{Mahalanobis}}(X(t), Y(t)) = \sqrt{(X-Y)^T S^{-1}(X-Y)}$$

The advantage of using Mahalanobis distance is that it takes into consideration the correlations, S, between the time series by which different patterns can be identified and analysed with respect to a based or reference point. However, it does not work well in our project and its calculation takes too long.

3.4.4 Pattern Model Representation (PMR)

PMR is based on a piecewise linear representation (PLR) and is effective at describing the tendency of time series. Then, the pattern distance can be calculated to measure the similarity of tendency. This method overcomes the problem of time series mismatch based on point distance. According to the numbers of series' segmentations, pattern distance has a multi-scale feature and can reflect different similarities with various bandwidths. Because normalization is unnecessary, the calculation consumption of pattern distance is low. Therefore, this might be a good measurement in our project.

4. Experiments

4.1. Software

- Coding: Bash, R
- Visualization: ggplot2, R
- Environment: CHTC

4.2. Implementation

First of all, we use kaggle's official API to download the dataset into CHTC. Then we clean the dataset to deal with the missing values and wrong values. Then we do the data processing jobs to standardize the data. After this, we write Bash scripts to divide the whole dataset into about 1000 folders with each folder containing 100 csv files. Then we write R file that can calculate the similarity between two time series.

The target pattern we select is Telsa stock before Jan 1st 2020. The figure 3 below shows the pattern (before the first dashed line). We write CHTC scripts to run 100 in parallel and find time series in the stock that has the highest similarity with our target pattern. After finding those stocks, we will compare their performance with S&P500 in the next one month time period and see if this method could find out good stocks.

4.3. CHTC Output

Our results shows that the Pearson Correlation Coefficient method runs fast and behaves well in this project. We will talk about it in the following section. In this setting, we submitted 1000 jobs with each job containing 100 csv files. Every job takes about approximately 1 minute to complete. The resource usage is shown below in Table 3:

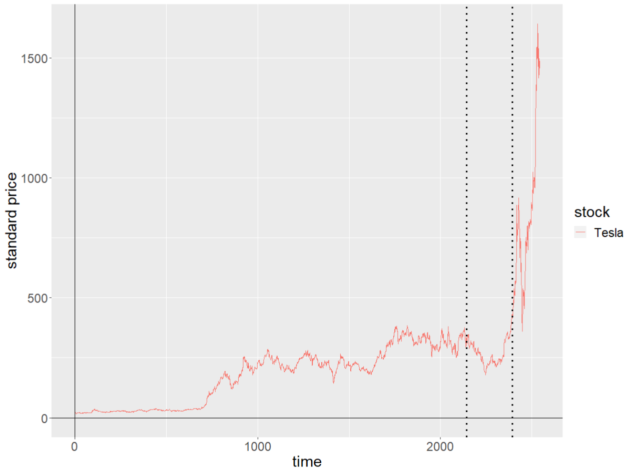

Figure 3. Tesla before 2019 as our pattern

Resources	Usage	Request	Allocated
CPU	1	1	1
Disk(KB)	152775	1048576	1327910
Memory(MB)	436	1024	1024

Table 3. Details of Computing Resource Usage

5. Results and Discussion

The result shows the top 5 stocks that has the highest similarities, i.e. the lowest distance are ROLLT.BO, C1S.BE, C1S.F, EPMT.JK, CEATLTD.DB. The following figure 4 shows their distance with our target pattern.

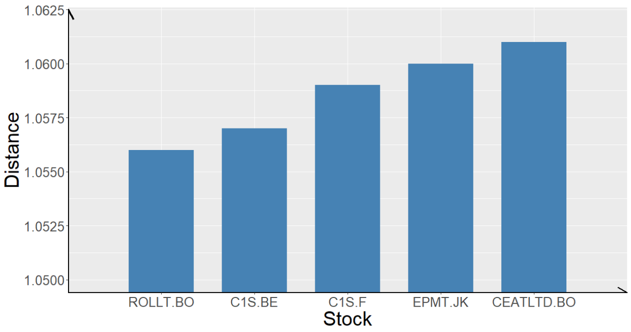

Figure 4. Top 5 stocks with their distances

By checking the stock chart of these stocks, we find these stocks having a extremely high similarities with our target pattern. Example graphs are shown below:

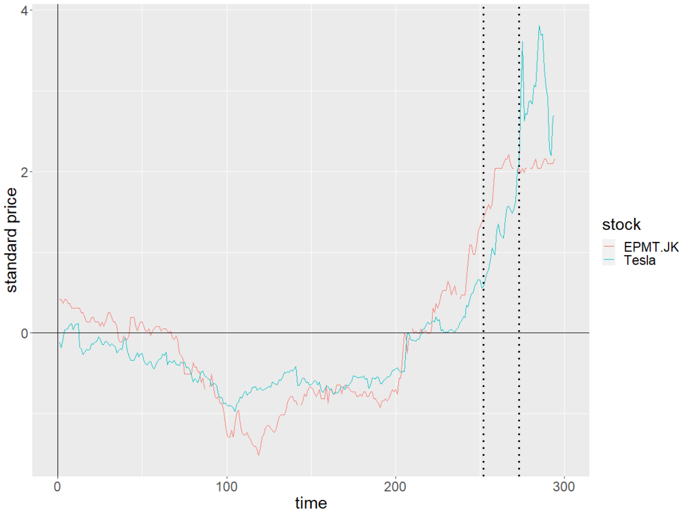

Figure 5. EPMT.JK Stock Chart

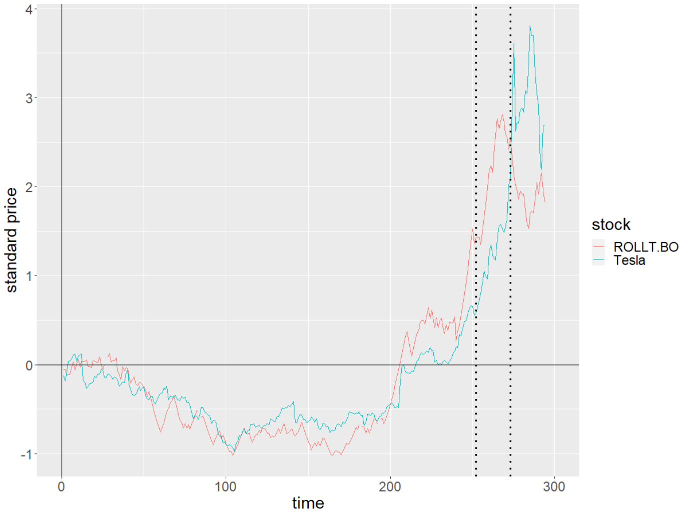

Figure 6. ROLLT.BO Stock Chart

Now it's time for us to verify if these stocks behave well. By checking the figure 5 and 6 above, we see that they all soar in the next year. But how about their performance compared with the whole market?

The figure 7 below shows the top 5 stock's performance comparing with the S&P500 index at their own time. It's obvious that their performance are much better than

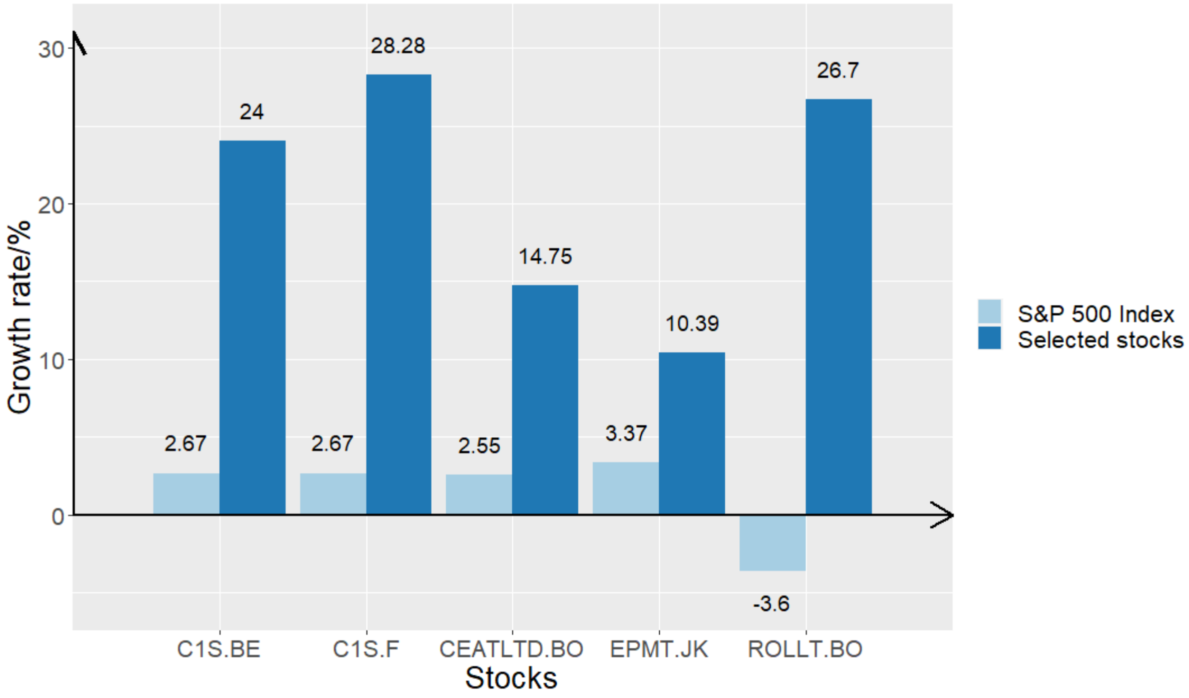

Figure 7. ROLLT.BO Stock Chart

S&P500 index at the same time period.

From the discussion above, we believe Technical Analysis Price Pattern does really work. If you choose Tesla before 2019 as a pattern and buying stocks that looks extremely like this pattern, the outcome is much better than buying the index like S&P500. In the future, if one stock chart has extremely high similarity with this pattern, it might have a better chance to become a good stock.

6. Conclusions

In this report, we presented our experiments to the Technical Analysis Price Pattern method. The method of simply looking at the stock that has some specific pattern may look strange, but our results based on the Kaggle dataset shows that some stock who has high similarities with another good stock pattern are more likely to become a good stock.As mentioned in previous section, currently, our results show that the Pearson Correlation Coefficient works best for now. But there are many measurements for time series data out there. The measurement of similarities on the stock dataset may be an interesting direction to go in future investigation.

7. Acknowledgements

The dataset of this project were obtain from Kaggle platform. We really appreciate that this online platform shares such great problems and real-industry datasets with public.

The computing resources in this project were provided by UW-Madison, Center for High Throughput Computing. We really appreciate the help from them.

We would also like to express our gratitude to Prof. Keith Levin, John Gillett and TA Bi Cheng Wu for their patient and detailed advice all the way through the project. Without their help and guidance, we can not finish this project with great results.

8. Appendix

For all the source codes, graphs and logs in our project, please check our GitHub repository at *Click me*.

References

[1] Adam Hayes (2020) *Introduction to Technical Analysis Price Patterns* `https://www.investopedia.com/articles/technical/112601.asp`.

[2] Kaggle (2020) *Kaggle is the world's largest data science community with powerful tools and resources to help you achieve your data science goals.* `https://www.kaggle.com/`

[3] A. Kianimajd,M. G. Ruano,P. Carvalho,J. HenriquesT.(2017) *Comparison of different methods of measuring similarity in physiologic time series.* `https://www.sciencedirect.com/science/article/pii/S2405896317333967/`

[4] Marcos Vinicius Naves Bedo(2013) *A Similarity-Based Approach for Financial Time Series Analysis and Forecasting* `https://link.springer.com/chapter/10.1007/978-3-642-40173-2_11`