A Data Preprocessing Method Applied to Cluster Analysis on Stock Data by Kmeans

Zhigang Xiong
Department of Computer Science and Engineering
Shanghai Jiao Tong University
Shanghai, China
Email: ben.bush@126.com

Assco. Prof. Zhongneng Zhang
Department of Computer Science and Engineering
Shanghai Jiao Tong University
Shanghai, China
Email: znzhang@sjtu.edu.cn

***Abstract*—Recent years, more and more data mining methods are involved in applications like stock price analysis or predication, etc. Kmeans is one commonly used algorithm in those applications. However, those applications only take the technical indices (indicators) as features of data, where may make some important information lost, like the cross of different curves formed by the same technical index with different parameters. In this paper, we propose one way to quantify the variation trend of different curves, which can make kmeans clustering algorithm more effective on stocks analysis.**

***Keywords*-clustering; kmens; stock; data process;**

I. Introduction

Since the stock market appeared, people have been studying and exploering its inner laws. In these years, there are many analytical methods emerged. Generally, we roughtly divide those analytical methods into two categories, the fundamental analysis and the technical analysis[1]. The fundamental analysis depends on macroeconomic indicators, such as the basic financial status of the company, money flow rate, inflation, etc. And also the fundamental analysis depends on some microeconomic indicators, such as the Listing Corporation's economic behavior and the corresponding economic variables[2]. Then the analyst make a selling or buying decision, after taking all these metrics into consideration. The technical analysis relies on the time series of stock price indices and various technical indicators. However the time series are often formed some complex patterns, which are discontinuous, sltatorial and not convergent. In this field, people usually analyze and make decisions depending on the statistical rules of time series data. Along with the development of data mining, some researchers try to introduce data mining methods into the field of technical analysis.

Clustering is a common method used in stock analysis. Researchers use cluster analysis to obtain the correlationship between different stock index[6], or to determine the period of the stock price[3]. Furthermore some researchers predict the future trend of stock price based on the cluster analysis, such as fuzzy rule based clustering prediction[4]. Among all clustering algorithms, kmeans is the most commonly used clustering algorithm in stock analysis, because the stock data are high dimensions[5], which is a huge challenge for other clustering methods.

However, researchers usually only take the technical indices (indicators) as features of data, which may make the interrelationship between different indices weakened. This means an important information will be lost: the intersection points of the curves formed by the time series of technical indicators. For example RSI (Relative Strength Index) is a famous technical indicator used in stock price trend analysis, which shows the power contrast between buyers and sellers within n days. Obviously, the 6 days RSI and 12 days RSI are two different time series, will form two different curves. In fact, the intersection points of that two curves may be the perfect deal points. More is people would like to concern the variation of distance between the 6 and 12 days RSI curves.

Hence, in order to solve above problem, we propose a data preprocessing method to quantify the variation of different curves. This proposed method will make the kmeans more effective. In this paper, Section I is an introduction, while in Section II the background knowledge and problem analysis are shown. The detail of our proposed method is placed in Section III. We also take an experiment to examine our proposed method, and its result are in Section IV. In the last section, we make a conclusion.

II. Background Knowledge & Problem Analysis

Cluster analysis are meaningful and useful. It divides data into clusters (groups) where the data share common characteristics. For this reason, cluster analysis make us more aware of the internal relationships of complex systems, like the variation trend of stock price which is a complex system. Some researchers use clustering algorithm to understand discreteness and corelationship of different stock[6]. The different stock in the same cluster indicate that they are very likely to have the same variation trend in the future. Also, clustering methods can be used to predict the stock price, by combing the fuzzy set theory[7]. In general, people are willing to use cluster analysis to guide the trading transaction of stocks.

The kmeans clustering technique is simple and commonly used. And kemans is a prototype-based partitional clustering technical which attempts to find the optimal k partitions. As it is prototype-based, the center of a cluster can be regard as the prototype of all points in such cluster. In other words, the features of a prototype represent the characteristics of all point in such cluster. One merit of kmeans clustering technical is, kmeans can be used for a wide variety of data types. And also kmeans is quite efficient, even though we should run multiple times to obtain a best solution, as it is not a stable algorithm. On the contrary, kmeans may meet a trouble when data contains outliers. However, kmeans is one of the few clustering algorithms that can deal with high dimensional data.

In order to facilitate analysis stock price, scientist create lots of technical indices, like RSI, WR, MA, etc. Generally, a single technical index can not accurately determine the future trend of a stock. In fact, we usually combine several indices to infer the volatility of stock prices. Different parameters of the same technical index also affect our judgement, for example 6 days and 12 days RSI extremely likely do not have the same value on the same day. Thus we should use both 6 and 12 days RSI to analyze the stock price. We can imagine, when using the clustering method to analyze stocks, the features will be a lot (a high dimensions dataset). Hence, kmeans is very suit for stock analysis.

$$SSE = \sum_{i=1}^{K} \sum_{x \in C_i} dist(c_i, x)^2 \tag{1}$$

When evaluate the validity of kmeans, SSE (sum of the squared error) is a good measurement (objective function). Equation 1 shows the defination of SSE, where C_i represent cluster i, c_i is the center of cluster i, $dist$ is the standard Euclidean (L_2) distance between two objects in Euclidean space. Indeed, the SSE represents the entropy of clusters. A smaller SSE means the points are more similiar in the same cluser, and the dataset are well separated by kmeans, where the smaller SSE the better. Thus, it's easily to know that the result with smaller SSE is better. Observing the definition of SSE, we notice that the different features of x are regarded as linear independence, which means the correlationship of different features of x are consider less in this expression. As we have mentioned in above section, the variation trend of two curves formed by the same technical index is the important basis for us to predict the future trend of stock price. In this point of view, a way to quantify the variation trend is desired.

III. Proposed preprocessing method

In the stock market analysis, most of the technical indicators have parameters. For example, RSI has a parameter n represent the size of watch window (days), which means we use recent n days trading data to calculate today's RSI. Usually, we take 12 or 14 as the window size for RSI. However, when conjecture the future trend of stock price, we would like to consider the difference between 6 days RSI and 12 RSI. Looking at the following three examples, each of them represents one real situation.

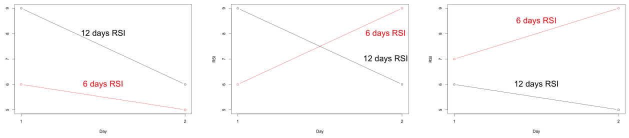

Figure 1. Relations of 6 and 12 days RSI

In Figure 1, the left subfigure shows that 6 days RSI gets close to 12 days RSI on day 2, the middle one shows an intersection is appeared on day 2, while the right subfigure shows that 6 days RSI gets separate on day 2. As we already know, RSI is used to measure the power contrast between buyers and sellers, the left subfigure of Figure 1 tells us that bayers power is weakening whether in 6 or 12 days, but the speed of decay is slowing down. Similarly, the middle subfigure tells us, the buyer power exceeds the seller on day2, and will continue to strengthen. The right subfigure tells us, in recent 6 days, the buyer power significantly stronger than it in 12 days, and the gap is widening. This is a sgin that the buyer confidence is recovering on day 2.

From the above analysis process we can see, the difference between 6 days RSI and 12 days RSI is quite important. And also, the varation trend of the difference is a key consideration, when analyze the stock data. Hence, we should not only take the time series of RSI as features, but also need to take the difference and the variation trend of difference between 6 and 12 days RSI, when using kmeans to anslyze the stocks. So we add two additional assessments, which can reflect the characteristics of the RSI curve better. The definition is shown below.

$$Dif = RSI(12) - RSI(6) \tag{2}$$

$$Trend(i) = (Dif(i) - Dif(i-1)) * sign(Dif(i-1)) \tag{3}$$

Equation 2 represent the difference between 6 and 12 days RSI. If $Dif(i) > 0$, it means last 12 days buyer power is greate than 6 days on day i, more is in recent 6 days, the desire to buy stocks is not strong enough. On the other hand, if $Dif(i) < 0$, it means recently people's desire to buy is more intense. The smaller the $Dif(i)$, the higher the enthusiasm of the purchase, in this case the market may become overbought, and vice versa.

In Equation 3, the $sign(x)$ is the sign function. And this equation represent the two RSI curves are close or far away, or cross on day i. If $Trend(i) > 0$, it means 6 days RSI gets close to 12 days RSI on day i, like the situation in left subfigure of Figure 1 where $Trend(2) = 2$. Specially, when $Trend(i)$ is relatively large and has a different sign with $Dif(i)$, we can deduce that there is a intersection point occurs on day i, and the 6 days RSI curve go through the 12 days RSI curve from bottom to top, like the example in middle subfigure of Figure 1. Otherwise, if $Trend(i) < 0$ and $Dif(i) > 0$, the 6 days RSI curve go through the 12 days RSI curve frorm top to bottom. In Figure 1, the right subfigure shows the separation of two curves, where the $Trend(2) = -5 < 0$.

Table I
TECHNICAL INDICES

Technical Index	Explanation
Open	The increase rate of today's opening price relative to the closing price of yesterday
Close	The increase rate of today's closing price relative to today's opening price
High	The increase rate of today's highest price relative to today's opening price
Low	The increase rate of today's lowest price relative to today's opening price
Volumn	The increase rate of today's trading volumn relative to the volumn of yesterday
Amount	The increase rate of today's trading amount relative to the amount of yesterday
WR	A technical analysis oscillator, its goals is try to tell us whether a stock is trading close to the high or the, or somewhere else, in recent n days
RSI	It's used to show the power constrast between buyers and sellers in recent n days
MA	The moving average of stock price of recent n days. Here we use its increase rate as the variable
BIAS	A measure of the degree of the stock price deviates from the average of n days. The stock price will tend to return to the average when BIAS is large
PSY	Psychological line is a kind of psychological tendency, which is based on the study of the investors. To tell us people are willing to buy or sell the stock in recent n days

We can see that above two equations can well describe the variation trend of two curves. Thus, we add Dif and $Trend$ as data features, when using kmeans to clustering stock data. In stock analysis, there are many technical indicators (indices) similar to RSI, people are willing to pay attention to the variation trend of two curves formed by different parameters, like WR, BIAS, KDJ and etc. Hence, before doing the cluster analysis, first we calculate the Dif and $Trend$ for those technical indices who require. And then put all Dif and $Trend$, together with technical indices, as the features of data. We believe that it will have a better result, after doing our proposed data preprocessing method. The date preprocessing flow is shown in Figure 2.

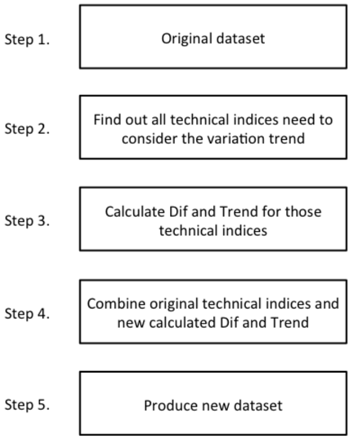
Step 1.
Original dataset
Step 2.
Find out all technical indices need to consider the variation trend
Step 3.
Calculate Dif and Trend for those technical indices
Step 4.
Combine original technical indices and new calculated Dif and Trend
Step 5.
Produce new dataset

Figure 2. Data Preprocessing Flow

IV. EXPERIMENTAL RESULTS

We take the time series data of daily CSI 300 index from 2007-01-04 to 2015-06-10 to examine our proposed data preprocessing method. And the technical indices used are shown in Table I. Among those indice, we calculate the Dif and $Trend$ for WR, RSI and BIAS, because people used to pay attention to their variation trend of two curves formed by different parameters.

To demonstrate the effectiveness of our approach, we prepare two datasets, one is the original dataset, and another one is the new dataset with taking the preprocessing method in Figure 2. Table II shows which features are included in which dataset, and the parameters used for each features. The "yes" in column "normal" means we normalize this feature (about the normalization, seeing [8], and the normalization function used is Equation 4.

$$x[i] = \frac{x[i] - min(x)}{max(x) - min(x)} \tag{4}$$

In order to observe how our proposed approach affect the results of kmeans, we take K from 2 to 25, as K is a user specified parameter in kmeans. And as everyone knows, kmeans is not a stable algorithm, the result of each run may be not the same. Thus, for each K, we should run serveral times to obtain a good result. In our experiment, for each K, we run 30 times.

As we mentioned in Section II, SSE (sum of the squared error) is a good measurement for kmeans, the smaller the SSE, the better the result we have. According to Equation 1, we understand that the more features we have, the larger SSE we may get. Thus, in this case, SSE does not work properly, because serveral extra features are added into the original dataset. To overcome this difficulty, other

Table II
DATASETS AND PARAMETERS

	parameter	normal	orig. data	new data
open		yes	yes	yes
close		yes	yes	yes
high		yes	yes	yes
low		yes	yes	yes
volumn		yes	yes	yes
amount		yes	yes	yes
wr	10	no	yes	yes
wr.dif	$wr(10) - wr(6)$	no	no	yes
wr.trend		yes	no	yes
rsi	12	no	yes	yes
rsi.dif	$rsi(12) - rsi(6)$	no	no	yes
rsi.trend		yes	no	yes
ma	10	yes	yes	yes
bias	6	yes	yes	yes
bias.dif	$bias(12) - bias(6)$	no	no	yes
bias.trend		yes	no	yes
psy	12	no	yes	yes

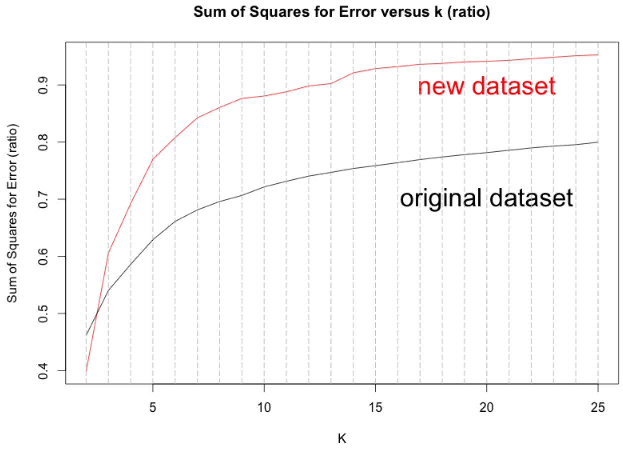

Figure 3. Experimental Result

measurement criteria should be used. Here we use following assessment to evaluate the results, seeing Equation 5.

$$SSE.ratio = \frac{SSE.between}{SSE.between + SSE.within} \tag{5}$$

The $SSE.within$ in Equation 5 is the SSE which we have mentioned in Section II. And $SSE.between$ is defined as Equation 6, which means the sum of the squared error between points in different clusters. It is not difficult to understand that the higher the $SSE.between$ of a clustering, the more separated the clusters are from one another, and the better the result we have. Thus, kmeans has a more validity clustering result, if $SSE.ratio$ is closer to 1, and vice versa.

$$SSE.between = \sum_{i=1}^{K} \sum_{x \in C_i} \sum_{j \neq i}^{K} (c_j, x) \tag{6}$$

Using the dataset in Table II, finally, we get the experimental result shown in Figure 3. We see that the curve of $SSE.ratio$ obtained by new dataset, which preprocess the data by our proposed method, is significantly over (closer to 1 from) the curve obtained by original dataset. Therefore, we can infer from this result that our approach is feasible and affective, which lead kmeans clustering algorithm to obtain a more volidity partitions.

V. CONCLUSION

There are some defects, if we use kmeans to make a cluster analysis on stock data in traditional way, because usually they only take the technical indices (indicators) as features of data, where some important information may lost, like the cross of different curves formed by the same technical index with different parameters. However, the variation trend of two curves formed by the same technical index with different parameters, including get close, far away and intersection. To compensate for this shortcoming, in this paper, we propose a data preprocessing method to quantify the variation trend, which introduce two extra assessments Dif and $Trend$, seeing Section II.

After taking an experiment, we observed, after using our proposed preprocessing method, kmeans clustering algorithm will obtain a more effective partitions, which is reflected in a significant increase on $SSE.ratio$. Therefore, we can say that our method is feasible and effective.

REFERENCES

[1] Edwards, Robert D., John Magee, and W. H. C. Bassetti. *Technical analysis of stock trends*. CRC Press, 2007.

[2] Boyer, M. Martin, and Didier Filion. *Common and fundamental factors in stock returns of Canadian oil and gas companies*. Energy Economics 29.3 (2007): 428-453.

[3] Arnott, Robert D. *Cluster analysis and stock price comovement*. Financial Analysts Journal 36.6 (1980): 56-62.s

[4] Lai, Robert K., et al. *Evolving and clustering fuzzy decision tree for financial time series data forecasting*. Expert Systems with Applications 36.2 (2009): 3761-3773.

[5] Donoho, David L. *High-dimensional data analysis: The curses and blessings of dimensionality*. AMS Math Challenges Lecture (2000): 1-32.

[6] Song, Dong-Ming, et al. *Evolution of worldwide stock markets, correlation structure, and correlation-based graphs*. Physical Review E 84.2 (2011): 026108.

[7] Liu, Chih-Feng, Chi-Yuan Yeh, and Shie-Jue Lee. *Application of type-2 neuro-fuzzy modeling in stock price prediction*. Applied Soft Computing 12.4 (2012): 1348-1358.

[8] Visalakshi, N. Karthikeyani, and K. Thangavel. *Impact of normalization in distributed k-means clustering*. international Journal of Soft computing 4.4 (2009): 168-172.