Journal of Theoretical and Applied Information Technology
30th September 2018. Vol.96. No 18

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

PREDICTING THE STOCK PRICE TRENDS USING A K-NEAREST NEIGHBORS-PROBABILISTIC MODEL

[1,2]LOCK SIEW HAN, [2]MD JAN NORDIN

[1]Universiti Kuala Lumpur, Malaysian Institute of Information Technology,

Kuala Lumpur, Malaysia

[2]Universiti Kebangsaan Malaysia, Faculty of Information Science and Technology, Bangi, Malaysia

E-mail: [1]lockshan@ymail.com, [2]jan@ukm.edu.my

ABSTRACT

This paper examines a hybrid model which combines a K-Nearest Neighbors (KNN) approach with a probabilistic method for the prediction of stock price trends. One of the main problems of KNN classification is the assumptions implied by distance functions. The assumptions focus on the nearest neighbors which are at the centroid of data points for test instances. This approach excludes the non-centric data points which can be statistically significant in the problem of predicting the stock price trends. For this it is necessary to construct an enhanced model that integrates KNN with a probabilistic method which utilizes both centric and non-centric data points in the computations of probabilities for the target instances. The embedded probabilistic method is derived from Bayes' theorem. The prediction outcome is based on a joint probability where the likelihood of the event of the nearest neighbors and the event of prior probability occurring together and at the same point in time where they are calculated. The proposed hybrid KNN-Probabilistic model was compared with the standard classifiers that include KNN, Naive Bayes, One Rule (OneR) and Zero Rule (ZeroR). The test results showed that the proposed model outperformed the standard classifiers which were used for the comparisons.

Keywords: *Stock Price Prediction, K-Nearest Neighbors, Bayes' Theorem, Naive Bayes, Probabilistic Method*

1. INTRODUCTION

Analyzing financial data in securities has been an important and challenging issue in the investment community. Stock price efficiency for public listed firms is difficult to achieve due to the opposing effects of information competition among major investors and the adverse selection costs imposed by their information advantage [5].

There are two main schools of thought in analyzing the financial markets. The first approach is known as fundamental analysis. The methodology used in fundamental analysis evaluates a stock by measuring its intrinsic value through qualitative and quantitative analysis. This approach examines a company's financial reports, management, industry, micro and macro-economic factors [1].

The second approach is known as technical analysis. The methodology used in technical analysis for forecasting the direction of prices is through the study of historical market data. Technical analysis uses a variety of charts to anticipate what are likely to happen. The stock charts include candlestick charts, line charts, bar charts, point and figure charts, OHLC (open-high-low-close) charts and mountain charts. The charts are viewable in different time frames with price and volume. There are many types of indicators used in the charts, including resistance, support, breakout, trending and momentum [2].

Several alternatives to approach this type of problem have been proposed, which range from traditional statistical modeling to methods based on computational intelligence and machine learning. Vanstone and Tan [3] surveyed the works in the domain of applying soft computing to financial trading and investment. They categorized the papers reviewed in the following areas: time series, optimization, hybrid methods, pattern recognition and classification. Within the context of financial trading discipline, the survey showed that most of the research was being conducted in the field of technical analysis. An integrated fundamental and technical analysis model was examined to evaluate the stock price trends by focusing on macro-

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

economic analysis. It also analyzed the company behavior and the associated industry in relation to the economy which in turn provide more information for investors in their investment decisions [4].

A nearest neighbor search (NNS) method produced an intended result by the use of KNN technique with technical analysis. This model applied technical analysis on stock market data which include historical price and trading volume. It applied technical indicators made up of stop loss, stop gain and RSI filters. The KNN algorithm part applied the distance function on the collected data. This model was compared with the buy-and-hold strategy by using the fundamental analysis approach [6].

Fast Library for Approximate Nearest Neighbors (FLANN) is used to perform the searches for choosing the best algorithm found to work best among a collection of algorithms in its library. Majhi et al. [7]-[8] examined the FLANN model to predict the S&P 500 indices, and the FLANN model was established by performing fast approximate nearest neighbor searches in high dimensional spaces.

Artificial neural networks (ANN) exhibit high generalization power as compared to conventional statistical tools. ANN is able to infer from historical data to identify the characteristics of performing stocks. The information is reflected in technical and financial variables. As a result, ANN is used as a statistical tool to explore the intricate relationships between the related financial and technical variables and the performance of stocks [9].

Neural network modeling can decode nonlinear regularities in asset price movements. Statistical inference and modifications to standard learning techniques prove useful in dealing with the salient features of economic data [10].

Some research has been carried out through the use of both qualitative and quantitative analysis. Shynkevich et al. [11] studied how the performance of a financial forecasting model was improved by the use of a concurrent, and appropriately weighted news articles, having different degrees of relevance to the target stock. The financial model supported the decision-making process of investors and traders. Textual pre-processing techniques were utilized for the predictive system. A multiple kernel learning technique was applied to predict the stock price movements. The technique integrated information extracted from multiple news categories while separate kernels were used to analyze each category. The news articles were partitioned according to some factors from the industries and their relevance to the target stock. The experiments were performed on stocks from the health care sector. The results showed that the financial forecasting model had achieved better performance when data sources contain increased categories of the number of relevant news. An enhanced model for this study incorporated additional data source using historical prices and made predictions based on both textual and time series data. Additional kernels can be employed for different data sources. The use of new categorical features was to improve the forecasting performance.

Linear regression is commonly used in financial analysis and forecasting. Many regression classifiers had demonstrated their usefulness to analyze quantitative data to make forecast by estimating the model parameters [12].

A regression driven fuzzy transform (RDFT) distributes a smoothing approximation of time series with a smaller delay as compared with moving average. This feature is important for forecasting tool where time plays a key role [33].

In high dimensional data, not all features are relevant and have an influence on the outputs. An Enhanced Feature Representation Based on Linear Regression Model for Stock Market Prediction was evaluated to investigate the statistical metrics used in feature selection that extracts the most relevant features to reduce the high dimensionality of the data. The statistical metrics include Information Gain, Term Frequency-Invert Document Frequency and the Document Frequency. The study illustrated that the identification of the relevant feature representations produced better result in the prediction output [31].

Volatility indicates the risk of a security. The Generalized Autoregressive Conditional Heteroskedasticity (GARCH) process is an approach used to estimate volatility in financial markets. The Seemingly Unrelated Regressions (SUR) is a generalization of a linear regression model that comprises several indicator relationships that are linked by the fact that their volatilities are correlated. A GARCH-SUR model was evaluated and demonstrated that the existence of a significant relationship between the volatility of macroeconomic variables and the stock market volatility in the financial markets [32].

As a company raises investment capital by offering its security to the public for the first time in an Initial Public Offering (IPO), there is a lot of volatilities in the absence of IPO lock-up period which is a cooling-off period that allows for the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

newly issued securities to stabilize without additional selling pressures from insiders. A study was conducted to investigate the lock-up provisions of IPOs and their effect on price changes around the lock-up expiry periods by analyzing the Efficient Market Hypothesis (EMH) in relation to the lock-up provision by using the standard Event Study Methodology and the Comparison Period Returns Approach (CPRA). The results showed that regardless of whether the market risks were incorporated in the analysis, the financial markets remained in semi-strong form around the lock-up expiry date [29]. The results of the CPRA and the event study methodology also showed a positive abnormal trading volume at lock-up expiry date for most of the sectors in the IPO market [30].

Ma et al. [13] proposed a hybrid financial time series model by combining Support Vector Regression (SVR), Trend model and Maximum Entropy (ME) based on ANN for forecasting trends in fund index. The study showed that the hybrid model extracted the financial features characteristics to formulate an improved predictive model.

A hybrid intelligent data mining methodology based on Genetic Algorithm - Support Vector Machine Model [14] was reviewed to explore stock market tendency. This approach makes use of the genetic algorithm for variable selection in order to improve the speed of support vector machine by reducing the model complexity, and then the historical data is used to identify stock market trends.

Hybrid techniques can be used to improve the existing forecasting models due to the limitation of ANN like black box technique [15]. A combination of methods such as fuzzy rule-based system [16]-[17], fuzzy neural network [18] and Kalman filter with hybrid neuro-fuzzy architecture [19] have been developed to predict financial time series data.

This research studies a hybrid approach through the use of KNN algorithm and a probabilistic method for predicting the stock price trends.

1.1 Comparison of various learning classifiers

When developing a classifier using various functions from different classifiers, it is important to compare the performances of the classifiers. Simulation results can provide us with direct comparison results for the classifiers with a statistical analysis of the objective functions. The hybrid KNN-Probabilistic model was compared with the supervised learning and classification algorithms, including KNN, Naïve Bayes, OneR and ZeroR.

1.1.1 K-Nearest Neighbors Classifier

The KNN algorithm is used to measure the distance between the given test instance and all the instances in the data set, this is done by choosing the *k* closest instances and then predict the class value based on these nearest neighbors. *k* is the assigned number of neighbors voting on the test instance. As such KNN is often referred to as case-based learning or an instance-based learning where each training instance is a case from the problem domain. KNN is also referred to as a lazy learning algorithm due to the fact that there is no learning of the model required and all of the computation works happen at the time a prediction is requested. KNN is a non-parametric machine learning algorithm as it makes no assumptions about the functional form of the problem being solved. Each prediction is made for a new instance (x) by searching through the entire training set for the *k* most nearest instances and applying majority voting rule to determine the prediction outcome [20].

A variety of distance functions are available in KNN which include Euclidean, Manhattan, Minkowski and Hamming. The Euclidean distance function is probably the most commonly used in any distance-based algorithm. It is defined as [21]

$$d(x,y) = \sqrt{\sum_{i=1}^{k} (x_i - y_i)^2} \quad (1)$$

where,

x and y are two data vectors and *k* is the number of attributes.

The Manhattan distance function is defined as [21]

$$d(x,y) = \sum_{i=1}^{k} |x_i - y_i| \quad (2)$$

The Minkowski distance function is defined as [21]

$$d(x,y) = \left(\sum_{i=1}^{k} (|x_i - y_i|)^q \right)^{1/q} \quad (3)$$

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The distance functions for Euclidean, Manhattan and Minkowski are used for numerical attributes.

The Hamming distance function is defined as [21]

$$d(x,y) = \sum_{i=1}^{k} |x_i - y_i|$$

$$x = y \Rightarrow D = 0$$
$$x \neq y \Rightarrow D = 1 \quad (4)$$

Hamming distance is usually used for categorical attributes. The Hamming distance between two data vectors is the number of attributes in which they differ [21].

1.1.2 Bayes' Theorem

The Bayes' theorem plays an important role in probabilistic learning and classification. The Bayesian classification represents a supervised learning method as well as a statistical method for classification. It has learning and classification methods based on probability theory. The Bayesian classification is named after Thomas Bayes (1702-1761), who proposed the Bayes' theorem. Bayes' theorem is often called Bayes' rule. The Bayes' rule uses prior probability of each category given no information about an item. Bayesian classification provides probabilistic methods where prior knowledge and observed data can be combined. It has a useful perspective for evaluating a variety of learning algorithms. The Bayesian classification calculates explicit probabilities for hypothesis and it is also robust to noise in input data. Given a hypothesis h and data D which bears on the hypothesis, Bayes' theorem is stated as [22]

$$P(h/D) = \frac{P(h/D)P(h)}{P(D)} \quad (5)$$

where,

P(D): independent probability of D

P(h): independent probability of h: prior probability

P(h|D): conditional probability of h given D: posterior probability

P(D|h): conditional probability of D given h: likelihood

a. Naïve Bayes Classifier

The Naïve Bayes classifier is a classification method based on Bayes' theorem with independence assumptions among predictors. A Naive Bayes classifier assumes that the presence of a particular attribute in a class is unrelated to the presence of any other attribute. A Naive Bayes model is easy to build, with no complicated iterative parameter estimation that makes it particularly useful for very large data sets. In spite of its simplicity, the Naïve Bayes classifier often does surprisingly well and is widely used due to the fact that it often outperforms other more sophisticated classification techniques. The Bayes' rule from (5) can also be expressed as [23]

$$P(h/D) = P(d_1|h) \times P(d_2|h) \times \ldots P(d_n|h) \times P(h) \quad (6)$$

where,

$h_1, h_2, \ldots, h_n$ be a set of mutually exclusive events that together form the sample space S.

D be any event from the same sample space, such that $P(D) > 0$.

b. Bayesian Network Classifier

A Bayesian network is a type of graph which is used to model events for probabilistic relationships among a set of random variables. This can then be used for inference. The graph is called a directed acyclic graph (DAG). DAG contains nodes and edges. A node is also called a vertex. The nodes of the graph represent random variables. Edges represent the connections between the nodes. If two nodes are connected by an edge, it has an associated probability that it will transmit from one node to the other. The graph is directed and does not contain any cycles. A directed graph has ordered pairs of vertices and it consists of a set of vertices and a set of arcs. In a DAG diagram, a vertex is represented by a circle with a label, and an edge is represented by an arrow or line extending from one vertex to another. Bayesian Network provides a representation of the joint probability distribution over the variables. A problem domain is modeled by a list of variables $X_{1, \ldots,} X_n$. Knowledge about the problem domain is represented by a joint probability $P(X_{1, \ldots,} X_n)$ [24].

A joint probability refers to the probability of more than one variable occurring together, such as the probability of event A and event B. Notation

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

for joint probability of two events takes the form P(A∩B) which denotes the probability of the intersection of A and B.

Joint probability for dependent events of event A and event B can be expressed as [25]

$$P(A \cap B) = P(A) \times P(B|A) \quad (7)$$

where,

"∩" denotes the point where A and B intersect.

P(A) is the probability of event A occurs

P(B |A) is the conditional probability of B given A

Joint probability for independent events of event A and event B can be expressed as [25]

$$P(A \cap B) = P(A) \times P(B) \quad (8)$$

In a joint distribution for independent variables, two discrete random variables X and Y are independent if the joint probability mass function conforms to [25]

$$P(X = x \text{ and } Y = y) = P(X = x) \times P(Y = y) \text{ for all } x \text{ and } y. \quad (9)$$

1.2 Related Works

A study on a KNN approach that made use of economic indicators and classification techniques to predict the stock price trends yielded considerable precision. Four indicators were identified and calculated based on the technical indicators and their formulas. The values of the indicators were normalized in the range between -1 and +1. Accuracy and F-measure were calculated to evaluate the performance of the model. Calculation of these evaluation measures required estimating Recall and Precision which were assessed from True Negative (TN), False Negative (FN), True Positive (TP) and False Positive (FP). The performance of the KNN model was improved by using the optimal value for the *k* parameter. The evaluation of the KNN model and its performance is illustrated in Table 1 [27].

Table 1: KNN Model Performance.

Measurement	K Parameter		
	25	45	70
Accuracy	0.8138	0.8059	0.8132
F-measure	0.8202	0.8135	0.8190

Another study examined a model named Integrated Multiple Linear Regression-One Rule Classification Model (Regression-OneR) that predicts the stock class outputs in classification form based on the initial predicted outputs in regression form. The regression classifiers were used to predict the outputs in continuous values as opposed to the classification classifiers which were used to predict the outputs in categorical values. As illustrated in Table 2, the result showed that the hybrid regression-classification approach produced better accuracy rate and lower Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) as compared with the model which incorporated only a standard classification algorithm [28].

Table 2: Prediction Results of Classifiers.

Classifier	Accuracy	MAE	RMSE
Regression-OneR	85.0746	0.1493	0.3863
OneR	71.6418	0.2836	0.5325
ZeroR	64.1791	0.4619	0.4805
J48 Decision Tree	82.0896	0.2121	0.3796
REP Tree	76.1194	0.2764	0.4207

2. MATERIALS AND METHODS

Data were collected from Bursa Malaysia which is the main stock exchange in Malaysia. The data sources are corporate annual reports which include income statements, cash flow and balance sheet. The features in the data set were formulated based on fundamental analysis. The features were financial ratios which are indicators of the companies' financial health. Table 3 and Table 4 illustrate the structure of data set 1 and data set 2. Data set 1 contains original data from heterogeneous sources with different data types used for currency values and financial ratios. All of the variables in data set 1 have numerical data type. The data in data set 1 was transformed into data set 2 with variables in categorical data type. The transformed data set provides a way to measure rankings of stock price performance in a standardized data format. The transformation process is shown in Figure 1.

Figure 1 illustrates that initially the numerical data from corporate reports were used as inputs. At this stage the data are raw facts. The raw data are then used for calculations of the financial ratios. The values in financial ratios are fractional type. At this stage the data are semantic but diverse. After that, the fractional data are then converted into standardized percentage values based on performance interpretations of the features. At this

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

stage the data are standardized. A ranking table is set up to group performance categories based on ranges of percentage values. The table is then used in data mapping to categorize the data from the percentage data format into the categorical data format. At this stage the data are interpretable.

2.1 Data Sets

The naming convention for the features in the data sets are closely resembled to the financial ratio terms which include debt/equity, asset turnover, cash flow and return on equity [26]. The original source of data in data set 1 contains numeric values which were then converted into categorical values in data set 2. Debt_Equity, Asset_Turnover, Cash_Flow, Return_on_Equity are the independent variables. The values of the independent variables are used to predict the value of the class variable named Price_Trend. The variable named Price in data set 1 contains numerical values which do not provide semantic interpretations of the stocks since various stocks have various prices. Whereas the variable named Price_Trend in data set 2 contains standardized categorical values which indicate positive or negative price trend.

Table 3: Data set 1.

Feature	Data Type
Debt_Equity	Numeric
Asset_Turnover	Numeric
Cash_Flow	Numeric
Return_On_Equity	Numeric
Price	Numeric

Table 4: Data set 2.

Feature	Data Type
Debt_Equity	Categorical
Asset_Turnover	Categorical
Cash_Flow	Categorical
Return_On_Equity	Categorical
Price_Trend	Categorical

Debt/equity ratio serves as measure for a company's financial leverage; it is calculated by dividing a company's total liabilities by its stockholders' equity.

Asset turnover reflects a company's sales revenue generated relative to the value of its assets. The asset turnover ratio indicates the efficiency with which a company is deploying its assets in generating revenue.

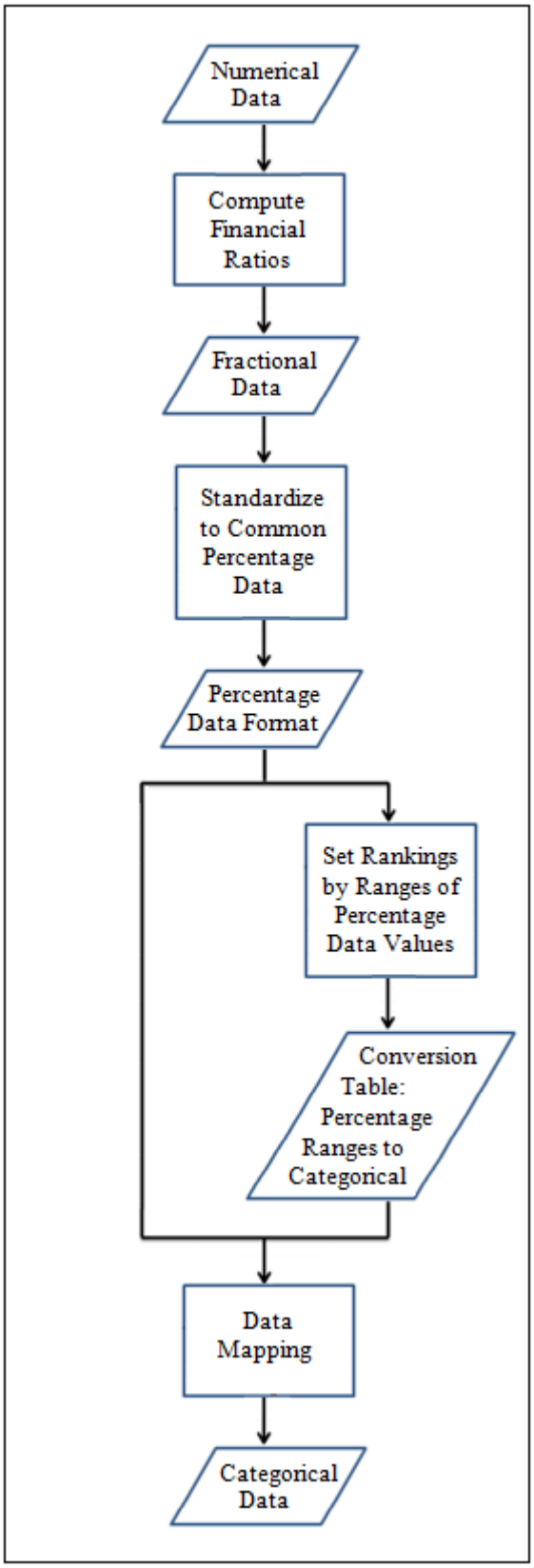

Figure 1: Flowchart of Data Transformation

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Cash flow is a measure of a company's financial health in terms of incomings and outgoings of cash, representing the operating activities of a company.

Return on equity is a measure of profitability based on how much profit a company generates with each dollar of stockholders' equity.

Price trend is the general direction and momentum of a market or of the price of security. If the price of security is going mainly upward, it is said to be on an upward price trend. The values of the dependable feature named "Price_Trend" consists of Profit and Loss labels. Profit indicates an upward price trend, whereas Loss indicates a downward price trend.

The data set 2 is structured to suit the approach of the KNN-Probabilistic model.

2.2 Methods

The flow chart in Figure 2 illustrates the process flow of the proposed model. Using the parallel approach, the model starts with computing the prior probabilities and the probabilities based on KNN approach simultaneously on both the Profit class and Loss class. KNN initialization process involves the use of the *k* value for the nearest neighbors of test instances. KNN then calculates the number of Profit class and Loss class instances based on the *k* number of nearest neighbors in the vicinity of each test instance. The outcome generated from KNN is then used by the probabilistic method for further classification.

The probabilistic method calculates the prior probabilities of Profit class and Loss class based on the number of instances in the data set. The outcome from the earlier KNN approach is used as an input as the probabilities of Profit class and Loss class by the nearest neighbors method. The joint probabilities of Profit class and Loss class can then be calculated using the outcomes of the prior probabilities and the calculated KNN's probabilities. Finally the predictive decision is made by comparing the joint probabilities of Profit class and Loss class.

The steps adopted for classification by KNN are illustrated as follows:

Steps:

- Classification: KNN
- Initialization of *k* value on nearest neighbors

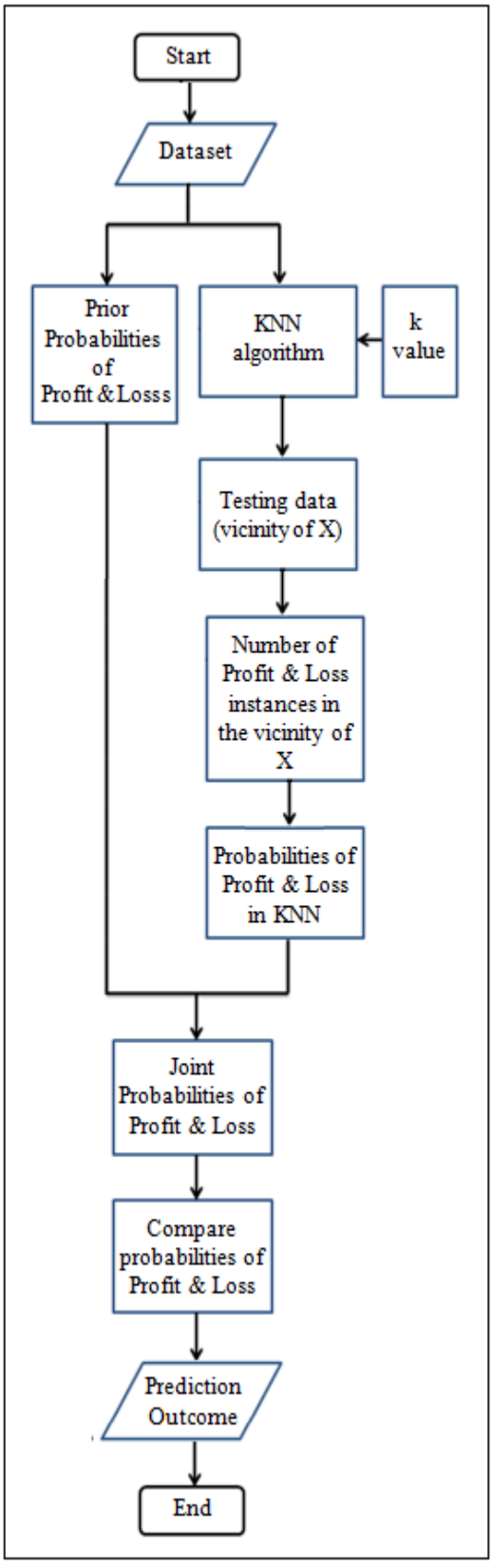

Figure 2: Flowchart of KNN-Probabilistic Model

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

- Compute the distance between the X query instance and all the training samples
- Sort the distance values
- Determine the nearest neighbors to the query instance based on the k value
- Calculate the number of Profit instances of the nearest neighbors in the vicinity of X query instance
- Calculate the number of Loss instances of the nearest neighbors in the vicinity of X query instance
- The steps adopted for classification by the probabilistic method is illustrated as follows:
- Steps:
- Classification: Probabilistic method
- Calculate the prior probabilities of Profit class and Loss class from the data set
- Calculate the KNN's probabilities of Profit class and Loss class based on the number of Profit nearest neighbors and the number of Loss nearest neighbors.
- Calculate the joint probabilities from the prior probabilities and KNN's probabilities on Profit class and Loss class
- Compare the joint probabilities of Profit class and Loss class
- Select the predictive value from the class values with the highest joint probability

3. RESULTS AND DISCUSSION

3.1 Results

The proposed model was tested and compared with four other standard algorithms, including KNN, Naïve Bayes, OneR and ZeroR. The test examined how accurate the tested algorithms predict the stock price trends, and evaluated the MAE and RMSE. Table 5 presents the test results.

The hybrid KNN-Probabilistic model has allowed us to achieve an estimated accuracy of 89.1725%, exceeding the stand alone KNN reported accuracy of 86.6667% and the Naive Bayes accuracy of 76.1194%. The accuracy rates for OneR and ZeroR classifiers were 71.6418% and 64.1791% respectively. KNN-Probabilistic model has MAE rate of 0.0667% and RMSE rate of 0.2582% which are much lower than the other classifiers.

Table 5: Prediction Results of Classifiers.

Classifier	Accuracy (%)	MAE	RMSE
KNN-Probabilistic	93.3333	0.0667	0.2582
KNN	86.6667	0.1333	0.3651
Naive Bayes	76.1194	0.1726	0.2824
OneR	71.6418	0.5325	0.6139
ZeroR	64.1791	0.4619	0.4805

Overall, KNN-Probabilistic model has better accuracy rate and error rates than the other classifiers used for comparisons. The test demonstrated that the hybrid mechanism of KNN and probabilistic method produced significantly improved results, compared with each of the KNN and Naïve Bayes classifiers.

3.2 Discussion

The proposed method begins with processing the data using data set 2, with each record contains a stock's financial features and the predicted outcomes in a structured categorical format. Using these records as inputs, stock price trends were predicted using the proposed hybrid KNN-Probabilistic model.

For KNN, the features in the data set are the data points in metric space with notion of distance. Each of the data set record contains a set of vectors and class labels associated with each vector. Each class label is either labeled as Profit for positive class or is labeled as Loss for negative class. The *k* value decides how many neighbors that can influence the classification. Initial step in KNN is to determine the appropriate *k* value. The *k* value is very training-data dependent. A small *k* value means that noise will have a higher influence on the result and a large value creates an overfit model. The use of k-fold cross-validation indicates the *k* value led to the highest classified generalizability. Typically odd number is used as *k* value when the number of classes is two, so that a decision can be determined based on the class value with the higher number of instances.

For KNN-Probabilistic model, an odd number *k* value is not required because a decision for prediction is not made at the initial stage of KNN classifier. A *k* value of even number is used to prevent unnecessary bias of unequal representations of the two classes at the stage of KNN method. A decision for prediction will be made based on the combined outcomes of the KNN

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

method and the probabilistic method. The KNN method determines the class instances that form the initial probabilities from the nearest neighbors' perspective. The probabilistic method makes use of a combination of probabilities in its decision making. When the inputs for prior probability and probability based on KNN are available, the predictive model can calculate the joint probability and make prediction on the class outcome.

The probabilistic model includes some functional relations between the unknown parameters and the observed data to allow us to make predictions. The goal of this statistical analysis is to estimate the unknown parameters in the proposed model. Initial stage includes identifying the optimal value for the *k* parameter. The computations used in the model include prior probability, probability in KNN and joint probability. The model has the following estimating computations.

The probability estimations based on KNN are,

The probability of Profit = The number of Profit instances of the nearest neighbors in the vicinity of the query instance / Number of *k* instances.

The probability of Loss = The number of Loss instances of the nearest neighbors in the vicinity of the query instance / Number of *k* instances.

The probabilities in KNN measures the support provided by the data for each possible value of the *k* parameter of KNN.

The computations of prior probability are,

The prior probability of Profit = Total number of Profit instances / Total number of all instances

The prior probability of Loss = Total number of Loss instances / Total number of all instances

The computations of joint probability are,

Joint probability of Profit = The probability of Profit based on KNN × The prior probability of Profit

Joint probability of Loss = The probability of Loss based on KNN × The prior probability of Loss

The steps of the process used for probability comparison is,

If Joint Probability of Profit > Joint Probability of Loss

Then prediction = Profit

Else

If Joint Probability of Loss > Joint Probability of Profit

Then prediction = Loss

Else

If Joint Probability of Profit == Joint Probability of Loss

Then repeat and re-adjust the *k* parameter.

4. CONCLUSION

The aim of this research is to improve the statistical fitness of the proposed model to overcome a KNN problem due to its computation approach. The KNN classifier can compute the empirical distribution over the Profit and Loss class values in the *k* number of nearest neighbors. However, the outcome is less than adequate due to sparse data. The KNN classifier has underfitting issue as it does not cater to generalization of sparse data outside the range of nearest neighborhood.

We have compared a hybrid KNN-Probabilistic model with four standard algorithms on the problem of predicting the stock price trends. Our results showed that the proposed KNN-Probabilistic model leads to significantly better results compared to the standard KNN algorithm and the other classification algorithms.

The limitation of the proposed model is that it applies a binary classification technique. The actual output of this binary classification model is a prediction score in two-class. The score indicates the model's certainty that the given observation belongs to either the Profit class or Loss class. For future work, the knowledge component is to transform the binary classification into multiclass classification. The multiclass classification involves observation and analysis of more than the existing two statistical class values. Additional research will include the application of the probabilistic model to multiclass data in order to provide more specific information of each class value. The newly formed multiclass classification will contain five class labels named "Sell", "Underperform", "Hold", "Outperform", and "Buy". In numerical values for mapping purpose, we will convert "Sell" to -2 which implies strongly unfavorable; "Underperform" to -1 which implies moderately unfavorable; "Hold" to 0 which implies neutral; "Outperform" to 1 which implies moderately favorable; and "Buy" to 2 which implies strongly favorable.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ACKNOWLEDGEMENT

This research is supported by the Malaysian Institute of Information Technology, Universiti Kuala Lumpur. The authors also would like to thank the Universiti Kebangsaan Malaysia (UKM) for financially supporting this research under Grant DIP-2016-018.

REFRENCES

[1] Benjamin Graham, Jason Zweig, and Warren E. Buffett, The Intelligent Investor, Publisher: Harper Collins Publishers Inc, 2003.

[2] Charles D. Kirkpatrick II and Julie R. Dahlquist, Technical Analysis: The Complete Resource for Financial Market Technicians (3rd Edition), Pearson Education, Inc., 2015.

[3] Bruce Vanstone and Clarence Tan, A Survey of the Application of Soft Computing to Investment and Financial Trading, Proceedings of the Australian and New Zealand Intelligent Information Systems Conference, Vol. 1, Issue 1, http://epublications.bond.edu.au/infotech_pubs/13/, 2003, pp. 211–216.

[4] Monica Tirea and Viorel Negru, Intelligent Stock Market Analysis System - A Fundamental and Macro-economical Analysis Approach, IEEE, 2014.

[5] Kian-Ping Lim, Chee-Wooi Hooy, Kwok-Boon Chang, and Robert Brooks, Foreign investors and stock price efficiency: Thresholds, underlying channels and investor heterogeneity, The North American Journal of Economics and Finance. Vol. 36, http://linkinghub.elsevier.com/retrieve/pii/S1062940815001230, 2016, pp. 1–28.

[6] Lamartine Almeida Teixeira and Adriano Lorena Inácio de Oliveira, A method for automatic stock trading combining technical analysis and nearest neighbor classification, Expert Systems with Applications, http://linkinghub.elsevier.com/retrieve/pii/S0957417410002149, 2010, pp. 6885–6890.

[7] Banshidhar Majhi, Hasan Shalabi, and Mowafak Fathi, FLANN Based Forecasting of S&P 500 Index. Information Technology Journal, Vol. 4, Issue 3, http://www.scialert.net/abstract/?doi=itj.2005.289.292, 2005, pp. 289–292.

[8] Ritanjali Majhi, G. Panda, and G. Sahoo, Development and performance evaluation of FLANN based model for forecasting of stock markets, Expert Systems with Applications, Vol. 36, Issue 3, http://linkinghub.elsevier.com/retrieve/pii/S0957417408005526, 2009, pp. 6800–6808.

[9] Tong-Seng Quah and Bobby Srinivasan, Improving returns on stock investment through neural network selection, Expert Systems with Applications, Vol. 17, Issue 4, http://linkinghub.elsevier.com/retrieve/pii/S095741749900041X, 1999, pp. 295–301.

[10] Halbert White, Economic prediction using neural networks: the case of IBM daily stock returns, IEEE International Conference on Neural Networks, http://ieeexplore.ieee.org/document/23959/, IEEE, 1988, pp. 451–458.

[11] Yauheniya Shynkevicha, T.M. McGinnity, Sonya A. Coleman, and Ammar Belatreche, Forecasting movements of health-care stock prices based on different categories of news articles using multiple kernel learning, Decision Support Systems, Vol. 85, http://linkinghub.elsevier.com/retrieve/pii/S0167923616300252, 2016, pp. 74–83.

[12] Han Lock Siew and Md Jan Nordin, Regression Techniques for the Prediction of Stock Price Trend, 2012 International Conference on Statistics in Science, Business and Engineering (ICSSBE), IEEE, 2012.

[13] Chi Ma, Junnan Liu, Hongyan Sun, and Haibin Jin, A hybrid financial time series model based on neural networks, IEEE, 2017.

[14] Lean Yu, Shouyang Wang, and Kin Keung Lai, Mining Stock Market Tendency Using GA-Based Support Vector Machines, Internet and Network Economics, http://link.springer.com/10.1007/11600930_33, 2005, pp. 336–345.

[15] Fu-Yuan Huang, Integration of an Improved Particle Swarm Algorithm and Fuzzy Neural Network for Shanghai Stock Market Prediction, 2008 Workshop on Power Electronics and Intelligent Transportation System.[Online].August 2008, IEEE, http://ieeexplore.ieee.org/document/4634852/, 2008, pp. 242–247.

[16] Ching-hsue Cheng, Tai-liang Chen, and Hia-jong Teoh, Multiple-Period Modified Fuzzy Time-Series for Forecasting TAIEX, Fourth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007), http://ieeexplore.ieee.org/document/4406190/, IEEE, 2007, pp. 2–6.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[17] Pei-Chann Chang, Chin-Yuan Fan, and Shih-Hsin Chen, Financial Time Series Data Forecasting by Wavelet and TSK Fuzzy Rule Based System, Fourth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007), http://ieeexplore.ieee.org/document/4406255/, IEEE, 2007, pp. 331–335.

[18] Lixin Yu and Yan-Qing Zhang, Evolutionary Fuzzy Neural Networks for Hybrid Financial Prediction, IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews), Vol. 35, Issue 2, http://ieeexplore.ieee.org/document/1424198/, IEEE, 2005, pp. 244–249.

[19] Chokri Slim, Neuro-fuzzy network based on extended Kalman filtering for financial time series, World Academy of Science, Engineering and Technology, Vol. 15, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.193.4464&rep=rep1&type=pdf, 2006, pp. 134–139.

[20] Gérard Biau and Luc Devroye, Lectures on the Nearest Neighbor Method, Springer, 2015.

[21] Saed Sayad, K Nearest Neighbors - Classification, [Online]. 2017, http://www.saedsayad.com/k_nearest_neighbors.htm, 2017, [Accessed: 10 January 2017].

[22] Dan Morris, Bayes' Theorem: A Visual Introduction for Beginners, Blue Windmill Media, 2017.

[23] James V Stone, Bayes' Rule With R: A Tutorial Introduction to Bayesian Analysis, Sebtel Press, 2016.

[24] Marco Scutari and Jean-Baptiste Denis, Bayesian Networks: With Examples in R, CRC Press, 2014.

[25] Peter Bruce and Andrew Bruce, Practical Statistics for Data Scientists: 50 Essential Concepts, O'Reilly Media, 2017.

[26] Ciaran Walsh, Key Management Ratios, 4th Edition (Financial Times Series), Prentice Hall, 2009.

[27] Seyed Enayatolah Alavi, Hasanali Sinaei, and Elham Afsharirad, Predict the Trend of Stock Prices Using Machine Learning Techniques, IAIEST, 2015.

[28] Lock Siew Han and Md Jan Nordin, Integrated Multiple Linear Regression-One Rule Classification Model for the Prediction of Stock Price Trend, Journal of Computer Sciences, Vol 13 (9), 2017, pp. 422-429.

[29] Abdolhossein Zameni and Othman Yong, Share Price Performance of Malaysian IPOs Around Lock-Up Expirations, Advanced Science Letters, Vol 23(9), 2017, pp. 8094-8102.

[30] Abdolhossein Zameni and Othman Yong, Lock-Up Expiry And Trading Volume Behaviour Of Malaysian Initial Public Offering's, International Journal of Economics and Financial Issues, Vol 6(3), 2016, pp. 12-21.

[31] Hani A.K. Ihlayyel, Nurfadhlina Mohd Sharef, Mohd Zakree Ahmed Nazri, and Azuraliza Abu Bakar, An Enhanced Feature Representation Based On Linear Regression Model For Stock Market Prediction, Intelligent Data Analysis, Vol 22(1), 2018, pp. 45-76.

[32] Lida Nikmanesh and Abu Hassan Shaari Mohd Nor, Macroeconomic Determinants of Stock Market Volatility: An Empirical Study of Malaysia and Indonesia, Asian Academy of Management Journal, Vol 21(1), 2016, pp. 161-180.

[33] Luigi Troiano, Pravesh Kriplani, and Irene Díaz, Regression Driven F-Transform and Application to Smoothing of Financial Time Series, IEEE, 2017.