**DATA MINING IN STOCK MARKET**

**By**

**ANANYA DATTA**

**ARCHIKAM NAGARAJ**

**CONTENTS**

**1: ABSTRACT**

**2: INTRODUCTION**

2.1: INTRODUCTION TO DATA MINING

2.2: ADVANTAGES OF DATA MINING

2.3: TECHNIQUE USED IN DATA MINING

**3: INTRODUCTION TO WEKA**

3.1: Introduction to WEKA

3.2: WEKA Tool Description

**4: PROBLEM DEFINITION**

4.1: Introduction to Application in Stock Market.

4.2: Introduction to Stock analysis.

4.3: Methodologies Used

4.4: Results from WEKA Tool

**5: CONCLUSION**

**6: REFERENCES**

**ABSTRACT**

Prediction of stock market trends has been an area of great interest both to those who wish to profit by trading stocks in the stock market and for researchers attempting to uncover the information hidden in the stock market data. Applications of data mining techniques for stock market prediction, is an area of research which has been receiving a lot of attention recently. Technical indicators are used in the present study to extract features from the historical SENSEX data. Apriori algorithm is then used to select best set of company to buy and sell from the generated rules. This study tries to help the investors in the stock market to decide the better Companies for buying or selling stocks based on the knowledge extracted from the historical prices, preferred by the experts, present price, high, low of such stocks.

**2. Introduction to Data Mining**

**2.1 Data Mining**

Data mining field at the intersection of computer science and statistics is the process that attempts to discover patterns in large data sets. It utilizes methods at the intersection of artificial intelligence, machine learning, statistics, and database systems.

The overall goal of the data mining process is to extract information from a data set and transform it into an understandable structure for further use. Aside from the raw analysis step, it involves database and data management aspects, data pre-processing model and inference considerations, interestingness metrics, complexity considerations, post-processing of discovered structures, visualization, and online updating.

Data mining uses sophisticated mathematical algorithms to segment the data and evaluate the probability of future events. Data mining is also known as Knowledge Discovery in Data (KDD).

Data Mining applies many older computational techniques from statistics, machine learning and pattern recognition

* Extract, transform, and load transaction data onto the data warehouse system.
* Store and manage the data in a multidimensional database system.
* Provide data access to business analysts and information technology professionals.
* Analyse the data by application software.
* Present the data in a useful format, such as a graph or table.
* The ultimate goal of data mining is prediction - and predictive data mining is the most common type of data mining and one that has the most direct business applications.

**2.2 Advantages of Data Mining**

1. Data mining is an important part of knowledge discovery process that analyzes large enormous set of data and gives us unknown, hidden and useful information and knowledge. Data mining has not only applied effectively in business environment but also in other fields such as weather forecast, medicine, transportation, healthcare, insurance, government and etc. Data mining brings a lot of advantages when using in a specific industry. Besides those advantages, data mining also has its own disadvantages as well such as privacy, security and misuse of information. We will examine the advantage of data mining in different industries in a greater detail.
2. Marking/Retailing: Data mining helps marketing companies to build models based on historical data to predict who will respond to new marketing campaign such as direct mail, online marketing campaign and etc. Through this prediction, marketers can have appropriate approach to sell profitable products to targeted customers with high satisfaction. Data mining brings a lot of benefit s to retail company in the same way as marketing. Through market basket analysis, the store can have an appropriate production arrangement in the way that customers can buy frequent buying products together with pleasant. In addition, it also help the retail company offers a certain discount for particular products what will attract customers.
3. Banking/Crediting: Data mining can assist financial institutions in areas such as credit reporting and loan information. Data mining gives financial institutions information about loan information and credit reporting. By building a model from previous customer’s data with common characteristics, the bank and financial can estimate what are the god and/or bad loans and its risk level. In addition, data mining can help banks to detect fraudulent credit card transaction to help credit card’s owner prevent their losses. For example, by examining previous customers with similar attributes, a bank can estimated the level of risk associated with each given loan.
4. Law enforcement: Data mining can aid law enforcers in identifying criminal suspects as well as apprehending these criminals by examining trends in location, crime type, habit, and other patterns of behaviours.
5. Researchers: Data mining can assist researchers by speeding up their data analyzing process; thus, allowing those more time to work on other projects.
6. Manufacturing: By applying data mining in operational engineering data, manufacturers can detect faulty equipments and determine optimal control parameters. For example semi-conductor manufacturers had a challenge that even the conditions of manufacturing environments at different wafer production plants are similar, the quality of wafer are lot the same and some for unknown reasons even contain defects. Data mining has been applied to determine the ranges of control parameters that lead to the production of golden wafer. Then those optimal control parameters are used to manufacture wafers with desired quality.

**2.3 Techniques used in Data Mining**

Various algorithms and techniques like Classification, Clustering, Regression, Artificial Intelligence, Neural Networks, Association Rules, Decision Trees, Genetic Algorithm, Nearest Neighbour method etc., are used for knowledge discovery from databases.

**2.3.1 Classification**

Classification is the most commonly applied data mining technique, which employs a set of pre-classified examples to develop a model that can classify the population of records at large. Fraud detection and credit risk applications are particularly well suited to this type of analysis. This approach frequently employs decision tree or neural network-based classification algorithms. The data classification process involves learning and classification. In Learning the training data are analyzed by classification algorithm.

**Types of classification models:**

1. **J48 Decision Tree Induction Algorithm**

The J48 algorithm gives several options related to tree pruning. Many algorithms attempt to "prune", or simplify, their results. Pruning produces fewer, more easily interpreted results. More importantly, pruning can be used as a tool to correct for potential over fitting. The basic algorithm described above recursively classifies until each leaf is pure, meaning that the data has been categorized as close to perfectly as possible. This process ensures maximum accuracy on the training data, but it may create excessive rules that only describe particular idiosyncrasies of that data. When tested on new data, the rules may be less effective. Pruning always reduces the accuracy of a model on training data. This is because pruning employs various means to relax the specificity of the decision tree, hopefully improving its performance on test data. The overall concept is to gradually generalize a decision tree until it gains a balance of flexibility and accuracy.

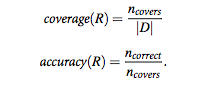
J48 employs two pruning methods. The first is known as sub tree replacement. This means that nodes in a decision tree may be replaced with a leaf -- basically reducing the number of tests along a certain path. This process starts from the leaves of the fully formed tree, and works backwards toward the root. The second type of pruning used in J48 is termed sub tree rising.

1. **Rule-based Classification**

The rule-based classiﬁers learned model is represented as a set of IF-THEN rules. An IF-THEN rule is an expression of the form:

**IF condition THEN conclusion.**

A rule R can be assessed by its coverage and accuracy. Given a tuple, X, from a classlabeled data set, D, let ncovers be the number of tuples covered by R; ncorrect be the number of tuples correctly classiﬁed by R; and |D| be the number of tuples in D. We can deﬁne coverage accuracy R as

[](https://sites.google.com/a/kingofat.com/wiki/data-mining/classification/Picture%205.png?attredirects)

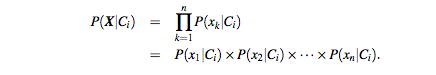
We can build a rule-based classiﬁer by extracting IF-THEN based classiﬁer by extracting based classiﬁer by extracting based classiﬁer by extracting IF rules from a decision tree. To extract rules from a decision tree, one rule is created for each path from the root to a leaf node. Each splitting criterion along a given path is logically ANDed to form the rule antecedent (“IF” part). The leaf node holds the class prediction, forming the rule consequent (“THEN” part). Then, the rule set should be pruned. There are assorted methods to do this.

**c) Bayesian Classification**

Bayesian classiﬁers are statistical classiﬁers. They can predict class membership probabilities, such as the probability that a given tuple belongs to a particular Class. Bayesian classiﬁcation is based on Bayes theorem; Studies comparing classiﬁcation algorithms have found a simple Bayesian classiﬁer known as the naive Bayesian classiﬁer to be comparable in performance with decision tree and selected neural network classiﬁers. Bayesian classiﬁers have also exhibited high accuracy and speed when applied to large databases.

Naïve Bayesian classiﬁers assume that the effect of an attribute value on a given class is independent of the values of the other attributes. This assumption is called class conditional independence. It is made to simplify the computations involved and, in this sense, is considered “naïve.” Bayesian belief networks are graphical models, which unlike naïve Bayesian classiﬁers allow the representation of dependencies among subsets of attributes. Bayesian belief networks can also be used for classiﬁcation.

So here the core formulation is



**d) Nearest-Neighbour Classiﬁers**

One popular example method is k-Nearest-Neighbour Classiﬁers. Nearest-neighbour classiﬁers are based on learning by analogy, that is, by comparing a given test tuple with training tuples that are similar to it. The training tuples are described by n attributes. Each tuple represents a point in an n-dimensional space. In this way, all of the training tuples are stored in an n-dimensional pattern space. When given an unknown tuple, a k-nearest-neighbour classiﬁer searches the pattern space for the k training tuples that are closest to the unknown tuple. These k training tuples are the k “nearest neighbours” of the unknown tuple. The unknown tuple is assigned the most common class among its k nearest neighbours.

Nearest-neighbour classiﬁers can also be used for prediction, that is, to return a real-valued prediction for a given unknown tuple. In this case, the classiﬁer returns the average value of the real-valued labels associated with the k nearest neighbour of the unknown tuple.

Nearest-neighbour classiﬁers use distance-based comparisons that intrinsically assign equal weight to each attribute. They therefore can suffer from poor accuracy when given noisy or irrelevant attributes. The method, however, has been modiﬁed to incorporate attribute weighting and the pruning of noisy data tuples. The choice of a distance metric can be critical. The Manhattan (city block) distance or other distance measurements may also be used. Nearest-neighbour classiﬁers can be extremely slow when classifying test tuples.

**e) Artificial Neural Network**

The word network in the term 'artificial neural network' refers to the inter–connections between the neurons in the different layers of each system. An example system has three layers. The first layer has input neurons, which send data via synapses to the second layer of neurons, and then via more synapses to the third layer of output neurons. More complex systems will have more layers of neurons with some having increased layers of input neurons and output neurons. The synapses store parameters called "weights" that manipulate the data in the calculations.

An ANN is typically defined by three types of parameters:

* The interconnection pattern between different layers of neurons
* The learning process for updating the weights of the interconnections
* The activation function that converts a neuron's weighted input to its output activation.

Training a neural network model essentially means selecting one model from the set of allowed models (or, in a Bayesian framework, determining a distribution over the set of allowed models) that minimizes the cost criterion. There are numerous algorithms available for training neural network models; most of them can be viewed as a straightforward application of optimization theory and statistical estimation.

Most of the algorithms used in training artificial neural networks employ some form of gradient descent. This is done by simply taking the derivative of the cost function with respect to the network parameters and then changing those parameters in a gradient-related direction.

**2.3.2 Association Analysis**

In data mining, association rule learning is a popular and well researched method for discovering interesting relations between variables in large databases. It is intended to identify strong rules discovered in databases using different measures of interestingness. Based on the concept of strong rules,. Introduced association rules for discovering regularities between products in large-scale transaction data recorded by point-of-sale (POS) systems in supermarkets. For example, the rule {onion, potatoes} => {burger} found in the sales data of a supermarket would indicate that if a customer buys onions and potatoes together, he or she is likely to also buy hamburger meat. Such information can be used as the basis for decisions about marketing activities such as, e.g., promotional pricing or product placements. In addition to the above example from market basket analysis association rules are employed today in many application areas including Web usage mining, intrusion detection and bioinformatics. As opposed to sequence mining, association rule learning typically does not consider the order of items either within a transaction or across transactions.

Association rules are usually required to satisfy a user-specified minimum support and a user-specified minimum confidence at the same time. Association rule generation is usually split up into two separate steps:

* First, minimum support is applied to find all frequent item sets in a database.
* Second, these frequent item sets and the minimum confidence constraint are used to form rules.

While the second step is straightforward, the first step needs more attention.

Many algorithms for generating association rules were presented over time.

Some well-known algorithms are Apriori, Éclat and FP-Growth, but they only do half the job, since they are algorithms for mining frequent item sets. Another step needs to be done after to generate rules from frequent item sets found in a database.

**2.3.3 Cluster analysis**

Cluster analysis or clustering is the task of grouping a set of objects in such a way that objects in the same group (called cluster) are more similar (in some sense or another) to each other than to those in other groups (clusters). It is a main task of explorative data mining, and a common technique for statistical data analysis used in many fields, including machine learning, pattern recognition, image analysis, information retrieval, and bioinformatics.

Cluster analysis itself is not one specific algorithm, but the general task to be solved. It can be achieved by various algorithms that differ significantly in their notion of what constitutes a cluster and how to efficiently find them. Popular notions of clusters include groups with low distances among the cluster members, dense areas of the data space, intervals or particular statistical distributions. Clustering can therefore be formulated as a multi-objective optimization problem. The appropriate clustering algorithm and parameter settings (including values such as the distance function to use, a density threshold or the number of expected clusters) depend on the individual data set and intended use of the results.

Besides the term clustering, there are a number of terms with similar meanings, including automatic classification, numerical taxonomy, bryology and typological analysis. The subtle differences are often in the usage of the results: while in data mining, the resulting groups are the matter of interest, in automatic classification primarily their discriminative power is of interest.

Clustering algorithms can be categorized based on their cluster model, as listed

* Connectivity based clustering (hierarchical clustering)
* k-means clustering
* Distribution-based clustering

**3 INTRODUCTION TO WEKA**

**3.1 WEKA**

The Weka workbench contains a collection of visualization tools and algorithms for data analysis and predictive modelling, together with graphical user interfaces for easy access to this functionality. The original non-Java version of Weka was a TCL/TK front-end to (mostly third-party) modelling algorithms implemented in other programming languages, plus data pre-processing utilities in C, and a Make file-based system for running machine learning experiments.

Advantages of Weka include:

* Free availability under the GNU General Public License
* Portability, since it is fully implemented in the Java programming language and thus runs on almost any modern computing platform
* A comprehensive collection of data pre-processing and modelling techniques
* Ease of use due to its graphical user interfaces

**3.2 WEKA Tool Description**

**3.2.1 Basic Concepts**

WEKA, formally called Waikato Environment for Knowledge Learning, is a computer program that was developed at the University of Waikato in New Zealand for the purpose of identifying information from raw data gathered from agricultural domains. WEKA supports many different standard data mining tasks such as data pre-processing, classification, clustering, regression, visualization and feature selection. The basic premise of the application is to utilize a computer application that can be trained to perform machine Learning capabilities and derive useful information in the form of trends and patterns. WEKA is an open source application that is freely available under the GNU general public license agreement. Originally written in C the WEKA application has been completely rewritten in Java and is compatible with almost every computing platform.

It is user friendly with a graphical interface that allows for quick set up and Operation. WEKA operates on the predication that the user data is available as a flat file or relation, this means that each data object is described by a fixed number of attributes that usually are of a specific type, normal alpha-numeric or numeric values. The WEKA application allows novice users a tool to identify hidden information from database and file systems with simple to use options and visual interfaces.

The GUI Chooser consists of four buttons—one for each of the four major Weka applications—and four menus.

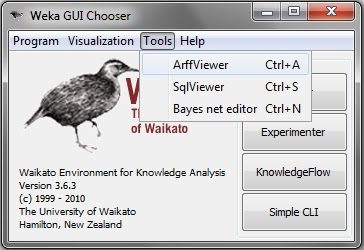
The buttons can be used to start the following applications:

* Explorer An environment for exploring data with WEKA (the rest of this documentation deals with this application in more detail).
* Experimenter An environment for performing experiments and conducting statistical tests between learning schemes.
* Knowledge Flow This environment supports essentially the same functions as the Explorer but with a drag-and-drop interface. One advantage is that it supports incremental learning.
* SimpleCLI Provides a simple command-line interface that allows direct execution of WEKA commands for operating systems that do not provide their own command line interface.

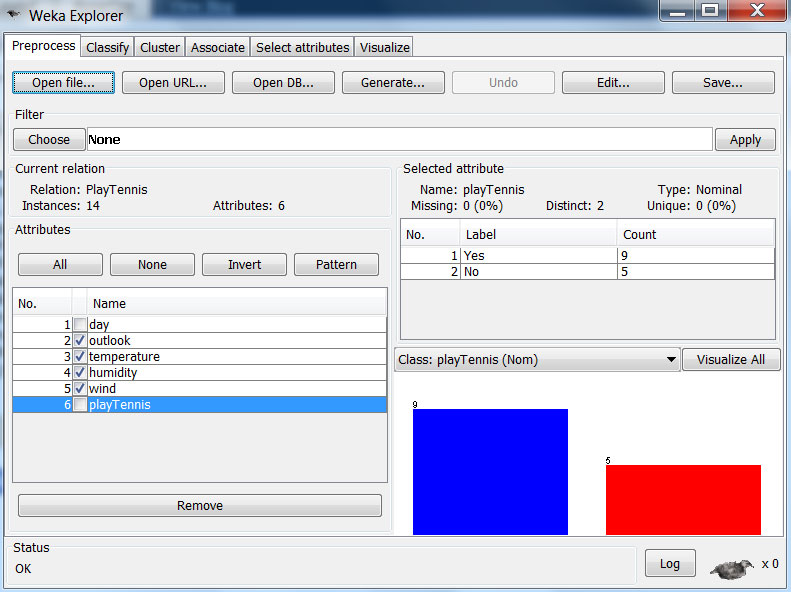
At the very top of the window, just below the title bar, is a row of tabs. When the Explorer is first started only the first tab is active; the others are greyed out. This is because it is necessary to open (and potentially pre-process) a data set before starting to explore the data.

The tabs are as follows:

1. Pre-process: Choose and modify the data being acted on.
2. Classify: Train and test learning schemes that classify or perform regression.
3. Cluster: Learn clusters for the data.
4. Associate: Learn association rules for the data.
5. Select attributes: Select the most relevant attributes in the data.
6. Visualize: View an interactive 2D plot of the data.



**Fig 3.2.1(a): Weka GUI Chooser**



**Fig 3.2.1(b): Weka Explorer**

**3.2.2 Dataset**

A set of data items, the dataset, is a very basic concept of machine learning. A dataset is roughly equivalent to a two-dimensional spread sheet or database table. In WEKA, it is implemented by the weka.core.Instances class. A dataset is a collection of examples, each one of class weka.core.Instance. Each Instance consists of a number of attributes, any of which can be nominal (= one of a predeﬁned list of values), numeric (= a real or integer number) or a string (= an arbitrary long list of characters, enclosed in “double quotes”).

**3.2.3 Classiﬁer**

Any learning algorithm in WEKA is derived from the abstract weka.classifiers.Classifierclass. Surprisingly little is needed for a basic classiﬁer: a routine which generates a classiﬁer model from a training dataset (= buildClassifier) and another routine which evaluates the generated model on an unseen test dataset (= classify Instance), or generates a probability distribution for all classes (= distributionForInstance).

A classiﬁer model is an arbitrary complex mapping from all-but-one dataset attributes to the class attribute. The speciﬁc form and creation of this mapping, or model, diﬀers from classiﬁer to classiﬁer. For example, ZeroR’s (=weka.classifiers.rules.ZeroR) model just consists of a single value: the most common class, or the median of all numeric values in case of predicting a numeric value (= regression learning). ZeroR is a trivial classiﬁer, but it gives a lower bound on the performance of a given dataset which should be signiﬁcantly improved by more complex classiﬁers. As such it is a reasonable test on how well the class can be predicted without considering the other attributes the simplest case is using a training set and a test set which are mutually independent. This is referred to as hold-out estimate. To estimate variance in these performance estimates, hold-out estimates may be computed by repeatedly resampling the same dataset – i.e. randomly reordering it and then splitting it into training and test sets with a speciﬁc proportion of the examples, collecting all estimates on test data and computing average and standard deviation of accuracy.

**4. PROBLEM DEFNITION**

**4.1. Introduction to Application in Stock Market.**

Over the past two decades many important changes have taken place in the environment of financial markets. The development of powerful communication and trading Facilities has enlarged the scope of selection for investors. Forecasting stock return is an important financial subject that has attracted researchers’ attention for many years. It involves an assumption that fundamental information publicly available in the past has some predictive relationships to the future stock returns.

**4.2. Introduction to Stock analysis.**

The stock analysis consists of the following six steps:

* Understanding the reason and objective of mining the stock prices.
* Understanding the collected data and how it is structured.
* Preparing the data that is used in the association model.
* Selecting the technique to build the model.
* Evaluating the model by using one of the well known evaluation methods.
* Deploying the model in the stock market to predict the best action to be taken, either selling or buying the stocks.
* Understanding the reason and objective of building the model

**4.3. Methodologies Used**

**Association**

An association rule has two parts, an antecedent (if) and a consequent (then). An antecedent is an item found in the data. A consequent is an item that is found in combination with the antecedent.

Association rules are created by analyzing data for frequent if/then patterns and using the criteria support and confidence to identify the most important relationships. Support is an indication of how frequently the items appear in the database. Confidence indicates the number of times the if/then statements have been found to be true.

In data mining, association rules are useful for analyzing and predicting customer behaviour. They play an important part in shopping basket data analysis, product clustering, catalogue design and store layout.

Programmers use association rules to build programs capable of machine learning.  Machine learning is a type of artificial intelligence (AI) that seeks to build programs with the ability to become more efficient without being explicitly programmed.

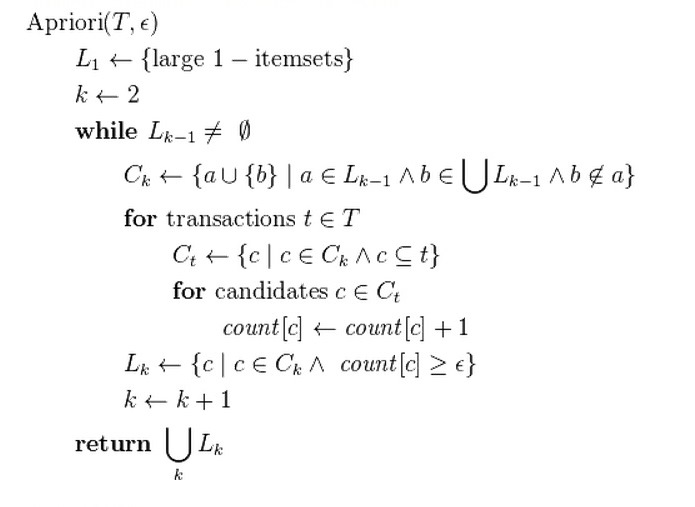
The strength of the association rule is quantified by the following factors:

• Confidence or predictability. A rule has confidence c if c% of the transactions in D that contain X also contain Y. A rule is said to hold on a dataset D if the confidence of the rule is greater than a user-specified threshold.

• Support or prevalence. The rule has support s in D if s% of the transactions in D contain both X and Y.

• Expected predictability. This is the frequency of occurrence of the item Y. So the difference between expected predictability and predictability (confidence) is a measure of the change in predictive power due to the presence of X. Usually, the algorithms only provide rules with support and confidence greater than the threshold values established.

**4.3.1. Aprori Algorithm**



Apriori is designed to operate on databases containing transactions (for example, collections of items bought by customers, or details of a website frequentation). Other algorithms are designed for finding association rules in data having no transactions, or having no timestamps (DNA sequencing). Each transaction is seen as a set of items (an item set). Given a thresholdC, the Apriori algorithm identifies the item sets which are subsets of at least C transactions in the database.

Apriori uses a "bottom up" approach, where frequent subsets are extended one item at a time (a step known as candidate generation), and groups of candidates are tested against the data. The algorithm terminates when no further successful extensions are found.

Apriori uses breadth-first search and a Hash tree structure to count candidate item sets efficiently. It generates candidate item sets of length k from item sets of lengthk-1. Then it prunes the candidates which have an infrequent sub pattern. The candidate set contains all frequent k-length item sets. After that, it scans the transaction database to determine frequent item sets among the candidates.

The pseudo code for the algorithm is given below for a transaction databaseT, and a support threshold of \epsilon. Usual set theoretic notation is employed, though note that T is a multi-set.  C_k Is the candidate set for levelk. At each step, the algorithm is assumed to generate the candidate sets from the large item sets of the preceding level, heeding the downward closure lemma. count[c] Accesses a field of the data structure that represents candidate set c, which is initially assumed to be zero. Many details are omitted below, usually the most important part of the implementation is the data structure used for storing the candidate sets, and counting their frequencies.

**4.3.2 Data Set**

**Before Pre- Processing**



**Pre-Processing**

1. Pre-processing Steps consists of removing unique element. Here company name is unique hence this is removed.
2. Present, high, low, Prev close are numeric values hence these are discretised. These discretised data renamed with appropriate range values.

**After Pre-Processing**

@relation st-weka.filters.unsupervised.attribute.Remove-R1-weka.filters.unsupervised.attribute.Discretize-B6-M-1.0-R1-4

@attribute 'PRESENT ' {0\_55,55\_101,101\_147,147\_193,193\_239,239\_285}

@attribute HIGH {0\_92,92\_172,172\_252,252\_333,333\_413,413\_495}

@attribute LOW {0\_44,44\_83,83\_122,122\_162,162\_201,201\_241}

@attribute PREV.CLOSE {0\_55,55\_101,101\_146,146\_192,192\_238,238\_284}

@attribute PREFERRED {sell, buy}

@data

239\_285,333\_413,162\_201,238\_284,sell

239\_285,333\_413,83\_122,238\_284,buy

239\_285,252\_333,44\_83,238\_284,buy

239\_285,252\_333,162\_201,238\_284,buy

239\_285,252\_333,44\_83,238\_284,buy

239\_285,252\_333,83\_122,238\_284,buy

239\_285,333\_413,83\_122,238\_284,buy

239\_285,252\_333,83\_122,192\_238,buy

239\_285,252\_333,122\_162,238\_284,buy

239\_285,252\_333,201\_241,238\_284,buy

239\_285,252\_333,122\_162,238\_284,buy

239\_285,252\_333,162\_201,192\_238,sell

193\_239,172\_252,44\_83,192\_238,sell

193\_239,333\_413,122\_162,238\_284,buy

193\_239,172\_252,122\_162,192\_238,sell

193\_239,172\_252,122\_162,192\_238,sell

193\_239,252\_333,44\_83,192\_238,buy

193\_239,333\_413,44\_83,192\_238,buy

193\_239,172\_252,44\_83,192\_238,sell

193\_239,252\_333,122\_162,192\_238,sell

193\_239,252\_333,0\_44,238\_284,sell

193\_239,172\_252,83\_122,192\_238,sell

193\_239,172\_252,83\_122,192\_238,sell

193\_239,252\_333,83\_122,192\_238,sell

193\_239,172\_252,201\_241,192\_238,sell

147\_193,172\_252,44\_83,146\_192,buy

147\_193,92\_172,83\_122,146\_192,buy

147\_193,172\_252,122\_162,146\_192,buy

147\_193,172\_252,44\_83,146\_192,buy

101\_147,172\_252,83\_122,101\_146,buy

101\_147,172\_252,83\_122,101\_146,buy

101\_147,92\_172,83\_122,101\_146,buy

101\_147,172\_252,122\_162,101\_146,buy

101\_147,92\_172,44\_83,101\_146,buy

101\_147,172\_252,0\_44,101\_146,sell

101\_147,92\_172,44\_83,101\_146,sell

101\_147,92\_172,44\_83,101\_146,buy

101\_147,172\_252,0\_44,101\_146,sell

101\_147,92\_172,0\_44,101\_146,sell

101\_147,172\_252,0\_44,101\_146,buy

101\_147,92\_172,83\_122,101\_146,buy

101\_147,92\_172,0\_44,101\_146,buy

101\_147,92\_172,44\_83,101\_146,buy

101\_147,92\_172,0\_44,101\_146,buy

101\_147,92\_172,83\_122,101\_146,buy

101\_147,92\_172,0\_44,101\_146,buy

101\_147,92\_172,0\_44,101\_146,buy

101\_147,92\_172,44\_83,101\_146,buy

101\_147,92\_172,44\_83,101\_146,sell

101\_147,92\_172,44\_83,101\_146,buy

101\_147,92\_172,83\_122,101\_146,sell

101\_147,92\_172,44\_83,101\_146,buy

101\_147,413\_495,83\_122,101\_146,buy

101\_147,92\_172,44\_83,101\_146,buy

101\_147,92\_172,0\_44,101\_146,buy

101\_147,92\_172,44\_83,101\_146,sell

101\_147,92\_172,44\_83,101\_146,buy

101\_147,92\_172,44\_83,55\_101,buy

101\_147,92\_172,83\_122,101\_146,buy

101\_147,92\_172,83\_122,101\_146,buy

101\_147,92\_172,44\_83,55\_101,sell

55\_101,92\_172,0\_44,55\_101,sell

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,44\_83,101\_146,sell

55\_101,172\_252,83\_122,55\_101,buy

55\_101,172\_252,83\_122,55\_101,buy

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,0\_44,55\_101,buy

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,44\_83,101\_146,sell

55\_101,172\_252,44\_83,55\_101,sell

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,44\_83,55\_101,sell

55\_101,92\_172,44\_83,55\_101,sell

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,0\_44,55\_101,buy

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,0\_44,55\_101,sell

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,0\_44,55\_101,buy

55\_101,92\_172,0\_44,55\_101,buy

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,0\_44,55\_101,buy

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,44\_83,55\_101,sell

55\_101,172\_252,83\_122,55\_101,buy

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,44\_83,55\_101,sell

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,0\_44,55\_101,sell

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,0\_44,55\_101,buy

55\_101,172\_252,83\_122,55\_101,buy

55\_101,92\_172,0\_44,55\_101,sell

55\_101,172\_252,83\_122,55\_101,buy

55\_101,92\_172,44\_83,55\_101,buy

55\_101,0\_92,83\_122,55\_101,buy

55\_101,0\_92,44\_83,55\_101,sell

55\_101,92\_172,44\_83,55\_101,sell

55\_101,92\_172,44\_83,55\_101,sell

55\_101,92\_172,0\_44,55\_101,sell

55\_101,92\_172,44\_83,55\_101,sell

55\_101,0\_92,44\_83,55\_101,sell

55\_101,92\_172,0\_44,55\_101,buy

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,0\_44,55\_101,sell

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,44\_83,55\_101,sell

55\_101,92\_172,0\_44,55\_101,buy

55\_101,92\_172,0\_44,55\_101,buy

55\_101,92\_172,44\_83,55\_101,sell

55\_101,92\_172,44\_83,55\_101,sell

55\_101,92\_172,44\_83,55\_101,sell

55\_101,92\_172,0\_44,55\_101,buy

55\_101,92\_172,44\_83,55\_101,sell

55\_101,92\_172,0\_44,55\_101,buy

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,0\_44,55\_101,buy

55\_101,92\_172,0\_44,55\_101,sell

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,44\_83,55\_101,sell

55\_101,92\_172,0\_44,55\_101,buy

55\_101,92\_172,44\_83,55\_101,sell

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,44\_83,55\_101,buy

55\_101,0\_92,0\_44,55\_101,sell

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,0\_44,55\_101,buy

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,0\_44,55\_101,sell

55\_101,0\_92,0\_44,55\_101,sell

55\_101,92\_172,44\_83,55\_101,sell

55\_101,0\_92,0\_44,55\_101,buy

55\_101,0\_92,0\_44,55\_101,buy

55\_101,0\_92,0\_44,55\_101,buy

55\_101,0\_92,0\_44,55\_101,buy

55\_101,0\_92,0\_44,55\_101,sell

55\_101,0\_92,44\_83,55\_101,sell

55\_101,92\_172,44\_83,55\_101,sell

55\_101,0\_92,0\_44,55\_101,buy

55\_101,0\_92,0\_44,55\_101,buy

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,44\_83,55\_101,buy

55\_101,92\_172,0\_44,55\_101,buy

55\_101,0\_92,0\_44,55\_101,sell

55\_101,0\_92,0\_44,55\_101,buy

55\_101,0\_92,44\_83,55\_101,buy

55\_101,0\_92,0\_44,55\_101,sell

55\_101,92\_172,44\_83,55\_101,buy

55\_101,0\_92,0\_44,55\_101,sell

55\_101,0\_92,0\_44,55\_101,sell

55\_101,0\_92,0\_44,55\_101,sell

55\_101,0\_92,44\_83,55\_101,sell

55\_101,92\_172,44\_83,55\_101,buy

55\_101,0\_92,44\_83,55\_101,buy

55\_101,0\_92,0\_44,55\_101,buy

55\_101,0\_92,0\_44,55\_101,buy

55\_101,0\_92,0\_44,55\_101,buy

55\_101,0\_92,0\_44,55\_101,buy

55\_101,0\_92,0\_44,55\_101,buy

55\_101,0\_92,44\_83,55\_101,buy

55\_101,0\_92,0\_44,55\_101,sell

55\_101,0\_92,0\_44,55\_101,buy

0\_55,0\_92,44\_83,0\_55,buy

0\_55,0\_92,0\_44,0\_55,buy

0\_55,0\_92,0\_44,0\_55,buy

0\_55,0\_92,0\_44,0\_55,sell

0\_55,0\_92,0\_44,0\_55,buy

0\_55,0\_92,0\_44,0\_55,buy

0\_55,0\_92,0\_44,0\_55,buy

0\_55,0\_92,0\_44,0\_55,buy

0\_55,0\_92,0\_44,0\_55,sell

0\_55,0\_92,0\_44,0\_55,buy

0\_55,0\_92,0\_44,0\_55,buy

**Attributes Used**

Present: Present value of the stock in the market.

High: Highest value reached in last 52 weeks in the market of the stock.

Low: Lowest value reached in last 52 weeks in the market of the stock.

Prev. Close: The previous closing price of the stock in the market.

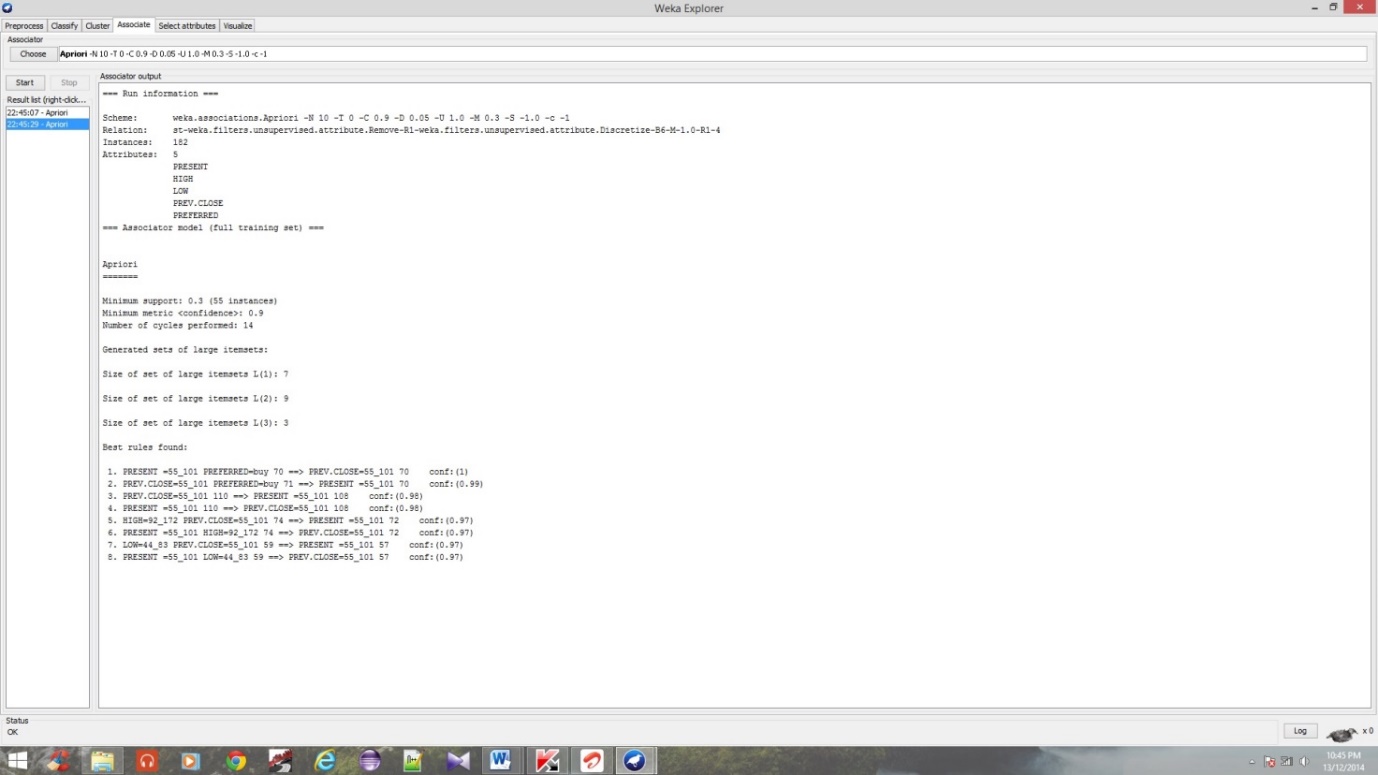
Preferred: Action (Buy or Sell) preferred by expert on stock.

**4.4. Results from WEKA Tool**

This shows that we have applied Apriori Algorithm to generate rules in our data by using the WEKA Tool as shown below:

Number of training instances: 182

Number of rules: 8



**Fig 4.3.3(a) Rules generated by apriori algorithm.**

=== Run information ===

Scheme: weka.associations.Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.3 -S -1.0 -c -1

Relation: st-weka.filters.unsupervised.attribute.Remove-R1-weka.filters.unsupervised.attribute.Discretize-B6-M-1.0-R1-4

Instances: 182

Attributes: 5

PRESENT

HIGH

LOW

PREV.CLOSE

PREFERRED

=== Associator model (full training set) ===

Apriori

=======

Minimum support: 0.3 (55 instances)

Minimum metric <confidence>: 0.9

Number of cycles performed: 14

Generated sets of large itemsets:

Size of set of large itemsets L(1): 7

Size of set of large itemsets L(2): 9

Size of set of large itemsets L(3): 3

Best rules found:

1. PRESENT =55\_101 PREFERRED=buy 70 ==> PREV.CLOSE=55\_101 70 conf:(1)

2. PREV.CLOSE=55\_101 PREFERRED=buy 71 ==> PRESENT =55\_101 70 conf:(0.99)

3. PREV.CLOSE=55\_101 110 ==> PRESENT =55\_101 108 conf:(0.98)

4. PRESENT =55\_101 110 ==> PREV.CLOSE=55\_101 108 conf:(0.98)

5. HIGH=92\_172 PREV.CLOSE=55\_101 74 ==> PRESENT =55\_101 72 conf:(0.97)

6. PRESENT =55\_101 HIGH=92\_172 74 ==> PREV.CLOSE=55\_101 72 conf:(0.97)

7. LOW=44\_83 PREV.CLOSE=55\_101 59 ==> PRESENT =55\_101 57 conf:(0.97)

8. PRESENT =55\_101 LOW=44\_83 59 ==> PREV.CLOSE=55\_101 57 conf:(0.97)

**5. CONCLUSION**

This study presents a proposal to use the association on the historical prices of the stocks to create decision rules that give buy or sell recommendations in the stock market. Such proposed model can be a helpful tool for the investors to take the right decision regarding their stocks based on the analysis of the historical prices of stocks in order to extract any predictive information from that historical data. The results for the proposed model were not perfect because many factors including but not limited to political events, general economic conditions, and investors’ expectations influence stock market.

**6. REFRENCE**

<http://www.lsi.us.es/redmidas/CEDI/papers/892.pdf>

<http://en.wikipedia.org/wiki/Apriori_algorithm>

http://www.money.rediff.com