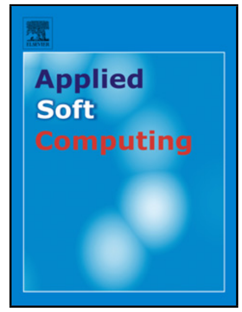
Accepted Manuscript



Title: Adaptive neuro fuzzy inference system for chart pattern matching in financial time series

Author: Yuqing Wan Yain-Whar Si

PII: S1568-4946(17)30147-3
DOI: http://dx.doi.org/doi:10.1016/j.asoc.2017.03.023
Reference: ASOC 4108

To appear in: *Applied Soft Computing*

Received date: 9-9-2016
Revised date: 13-2-2017
Accepted date: 13-3-2017

Please cite this article as: Yuqing Wan, Yain-Whar Si, Adaptive neuro fuzzy inference system for chart pattern matching in financial time series, *<![CDATA[Applied Soft Computing Journal]]>* (2017), http://dx.doi.org/10.1016/j.asoc.2017.03.023

Adaptive neuro fuzzy inference system for chart pattern matching in financial time series

Yuqing Wan[a], Yain-Whar Si[a,1,*]

[a] *Department of Computer and Information Science, University of Macau, Macau, China*

Abstract

In technical analysis, the appearance of chart patterns in financial time series is considered as one of the crucial signals in predicting future price trend. In recent years, various classification methods have been proposed by researchers to locate and identify potential chart patterns from input time series. This paper presents a novel application of adaptive neuro fuzzy inference system (ANFIS) for chart pattern matching in financial time series. The construction of ANFIS for chart patterns is described in the paper. In addition, we propose a method to determine the thresholds to implement chart patterns matching with the trained ANFIS model. The effectiveness and efficiency of the ANFIS model are compared with six pattern matching approaches on both synthetic datasets and real datasets. Experimental results reveal that the ANFIS model is effective in classifying different chart patterns when compared with other pattern matching approaches.

Keywords: chart patterns, financial time series, pattern matching, adaptive neuro fuzzy inference system

1. Introduction

Technical analysis is a methodology of predicting future price trend based on historical price or volume data. In technical analysis, the occurrence of some of the recurrent patterns within a financial time series is often considered as a signal in forecasting the price movement. These recurrent patterns are called "chart patterns" and they have been extensively studied by traders and stock market experts ([1], [2]). One of the crucial problems in technical analysis is how to

*Corresponding author

Email addresses: qinggor@qq.com (Yuqing Wan), fstasp@umac.mo (Yain-Whar Si)

[1] Faculty of Science and Technology, University of Macau, E11-4022, Avenida da Universidade, Taipa, Macau, China, Tel.: +853 8822-4454

locate and identify chart patterns in an efficient and accurate way. The complexity of this problem is further compounded by the shape similarity in some of the patterns and sheer amount of data that need to be processed in real-time for analysis.

Many pattern matching approaches have been proposed in recent years. Template-based (TB) [3] approach calculates the amplitude distance and temporal distance between a template of chart pattern and a sequence. Rule-based (RB) [3] approach identifies patterns by checking if the sequence complies with the predefined rules. TB can be considered as a "rigid approach" because the patterns recognized should be similar to the pattern templates to some extent. The similarity is confirmed based on a threshold. RB can also be considered as a rigid approach since rules used for matching are fixed in advance and only accept exact matches. Some approaches based on similarity measure of time series, such as dynamic time warping (DTW) [4] and Euclidean distance (ED), can be used to calculate the distance between the template and the sequence. ED calculates the point-to-point Euclidean distance between two sequences while DTW measures similarity of two sequences that are out of phase. One of the problems of DTW is that DTW focuses only on deformations of a pattern in the aspect of the elasticity in time. Support vector machine (SVM) is a well-known method for pattern classification and prediction in time series. In SVM, a maximum-marginal hyperplane to separate two different patterns is constructed in a high dimensional space. SVM is a linear binary classifier. When SVM is applied in the multiple classes classification, the dominant approach is to construct multiple binary classifiers. The cost of time and space are expensive when the number of classes is high. The HSMM has been applied in many different areas and is commonly used in speech recognition, speech synthesis, human activity recognition and handwriting recognition. The Hidden Semi-Markov Model (HSMM) is also used to detect specific waveforms in time series in [5] and [6]. An expectation-maximisation (EM) [7] algorithm is used to estimate the parameters in an HSMM. In EM, the forward-backward procedure [8] is used to calculate the re-estimation formulas and sum probability.

Adaptive-neuro fuzzy inference system (ANFIS) [9] combines the fuzzy logic and neural network. ANFIS acquires fuzzy rules from the given training dataset that has input-output data pairs. Fuzzy inference system uses fuzzy if-then rules to represent the characteristics of the training datasets. The neural network structure embedding in the fuzzy inference system identifies parameters with a learning algorithm. The primary advantages of ANFIS are the non-linearity and structured knowledge representation. In ANFIS, the distribution of the inputs are characterized by membership functions which describe the deformations of the chart pattern templates. ANFIS can be easily adopted for multiple classes classification problem by assigning different labels to different classes. ANFIS has been applied in the tasks of classification and prediction for automatic control, signal processing etc. In this paper, we compare the effectiveness and efficiency of five pattern matching approaches (TB, RB, ED, DTW, HSMM and SVM) with the ANFIS model on synthetic datasets and real datasets. The contributions of this paper can be summarized as follows:

- A novel application of ANFIS for chart patterns matching in financial time series is presented. To the best of authors' knowledge, there is no reported work on the application of ANFIS in classifying chart patterns from financial time series.
- To take into account all known patterns, we classify the 53 chart patterns into five categories according to their characteristics. Based on this classification, an ANFIS model for three representative patterns from the first three categories is constructed and tested. To the best of authors' knowledge, no one has ever considered the classification of all known chart patterns which are constructed from points. By using representative patterns, the proposed ANFIS model can be easily extended to classify any other chart patterns from the same category.
- A novel approach for using thresholds to classify different chart pattern with ANFIS model is presented in this paper. Thresholds denotes the degree of satisfying the rules which are used to recognize a chart pattern. In contrast to previous applications of ANFIS, the proposed approach not only use the final output of ANFIS, but also apply the thresholds to classify a new input sequence. As a result, the proposed approach achieves better results in excluding negative cases.
- Experiments on the comparison of the efficiency and effectiveness between the ANFIS model and the other six pattern matching show that ANFIS model is comparable with the SVM approach.

In section 2, we review the related work. Section 3 describes how to categorize 53 chart patterns. Section 4 briefly reviews the current pattern matching approaches. The adaptive neuro fuzzy inference system (ANFIS) is introduced in Section 5. In section 6, we conduct extensive experiments on the comparison of different pattern matching approaches on synthetic datasets and real datasets. Section 7 concludes the paper.

2. Related work

Adaptive neuro fuzzy inference system proposed by Jang [9] is a fuzzy inference system embedded in the framework of adaptive networks. ANFIS learns fuzzy if-then rules from the training datasets with the architecture of adaptive networks and the learning algorithm, therefore it can be widely used for classification and prediction. There are three types of fuzzy inference models: Mamdani fuzzy model, Tsukamoto fuzzy model and Sugeno fuzzy model [9]. The difference between these three models are the different ways of defuzzification or the way of calculating the output. For the purpose of efficiency, Sugeno type fuzzy model is adopted in this paper. The output of each rule in Sugeno type fuzzy model is the linear combination of input values plus a constant term and the final output is the weighted average of each rule's output. Grid partition and scatter partition are two common methods to partition the input space into a number of local fuzzy regions [10]. Grid partition has the problem of cures of dimensionality when the number of inputs is moderately large. Scatter partition using subtractive clustering technique [11] is also adopted to partition the input space in this paper. ANFIS can automatically find the fuzzy rules to represent the trained dataset thereby it can be used in matching of chart patterns.

ANFIS has been applied in the field of automatic control and signal processing. Mbede et al. [12] present the design of a neuro-fuzzy controller for mobile manipulator. The controller and modified Elman neural network is integrated to deal with structured and unstructured uncertainties. Melin et al. [13] use the ANFIS methodology to build a Sugeno fuzzy model for controlling the speed of a stepping motor drive. ANFIS is also applied in the field of biomedical research. Belal et al. [14] classify plethysmogram pulses into two categories with the ANFIS model. Feature extraction is performed before the training of the ANFIS model. In [15], six ANFIS classifiers are trained to detect the erythemato-squamous disease with features defining the disease indications as inputs. A combined ANFIS is trained by using the output of six ANFIS classifiers to improve the accuracy. The application of ANFIS model for classification of electroencephalogram (ECG) signals is presented by Güler et al. ([16], [17]). Feature extraction using wavelet transformation is performed on the ECG signals and then an ANFIS is trained with the features with the hybrid learning algorithm which is the combination of back propagation gradient descend and least squares method.

The ANFIS model is usually trained with the datasets of features of certain classes. For chart patterns in financial time series, the ANFIS model is trained using the features of the chart patterns.

These features are important points of the pre-defined chart pattern templates. In this paper, synthetic datasets of features of different chart patterns are used to train and test the ANFIS model. For testing with Hang Seng Index (HSI), feature extraction prepares the inputs for the trained ANFIS model by pre-processing the real dataset. The feature extraction is the segmentation of the original sequence to reduce the number of data points of the sequence, so that the segmented sequence has the same length as the defined chart pattern template or the length of the segmented sequence is the same as the number of inputs of the trained ANFIS model. Segmentation methods include the perceptually important points (PIP) method [18], the piecewise aggregate approximation (PAA) method [19], the piecewise linear approximation (PLA) method [20] and the turning points (TP) method [21]. The effect of different segmentation methods in financial time series is evaluated in [22]. PIP approach has a better performance than other segmentation methods. In the experiments, PIP is adopted as a pre-processing step for different pattern matching approaches to extract important points from the sequence.

In this paper, the ANFIS model is compared with the other pattern matching approaches. Recall that TB [3] and RB [3] pattern matching approaches are two rigid approaches. TB approach measures the similarity between the chart pattern template and the sequence by calculating their amplitude distance and temporal distance. A threshold needs to be determined as the condition of accepting a sequence as a chart pattern. TB approach matches sequences which are similar to the template in the vertical direction (amplitude distance) and the horizontal direction (temporal distance), therefore there is a possibility that the matched sequence appears deformed with respect to the template. The RB approach uses a set of predefined rules to identify the given sequence. Setting rules is subjective and therefore can affect the pattern matching results. ED approach is a simple similarity measurement whereas DTW [4] approach is a popular similarity measurement with many variations. TB, ED and DTW approaches calculate the distance between the chart pattern template and the sequence in different ways. TB approach calculates the similarity in two different directions (vertical and horizontal). ED approach calculates the point-to-point distance in vertical direction while DTW approach calculates the similarity in vertical direction in a warped way. For ED and DTW, when the calculated distance is less than a threshold, the sequence is accepted as a chart pattern.

SVM is a popular approach used in classification and prediction of time series. SVM is a supervised learning model with associated learning algorithms that can analyse data and recognise patterns. SVM is a non-probabilistic binary linear classifier that assigns a new input sequence into one category or the other. A kernel function is used to map the low dimensionality feature space to a high dimensionality feature space. Choosing a kernel function and speeding up the computation of the kernel matrix comprise two challenges in the study of the SVM [23]. In this paper, LIBSVM [24] is used in the experiments. LIBSVM is an open source machine learning library that implements the sequential minimal optimisation (SMO) algorithm for kernel SVMs. When LIBSVM is applied in classifying multiple classes, multiple binary classifiers are constructed. An instance is assigned to one of the two classes by every binary classifier. The class of the most number of assignment determines the classification of the instance. This method is also known as “one-versus-one”.

A hidden Markov model (HMM) is a doubly stochastic process [25]. The underlying stochastic process is a discrete time finite-state homogeneous Markov chain. An HMM has a hidden state sequence and an observable sequence of observations that are influenced by the hidden states. The hidden semi-Markov model (HSMM), an extension of the HMM, is intended to overcome the limitations of the HMM in some applications. The HSMM has been applied in many different areas and is commonly used in speech recognition, speech synthesis, human activity recognition and handwriting recognition. The HSMM is also used to detect specific waveforms in time series in [5] and [6]. In addition, an expectation-maximisation (EM) [7] algorithm is used to estimate the parameters in HSMM.

Recurrent neural networks (RNNs) [26] are popular models for tasks that involve time series. The hidden states of the network store information about the past elements of the sequence for later use. One of the training approaches of RNNs is long-short term memory (LSTM) [27] which is designed to avoid the long-term dependency problem. LSTM networks compute the hidden state in a different way than the standard RNNs which use logistic functions. RNNs have been widely used to solve many natural language processing problems, such as machine translation [28] and language modeling [29].

There are also other representation approach of time series in frequency domain. One of the transformation techniques is the discrete Fourier transforms (DFT). The discrete waveform transform (DWT) has been found to be more effective than DFT [30]. Discrete Wavelet Transformations use a set of functions called wavelets to represent a time series by decomposing the data into different components. Haar wavelet transform [31] is a kind of DWT which uses Haar wavelet to represent a time series. Given a time series, Haar wavelet transform decomposes the data into two components; "averages" and "differences". The component "averages" is calculated by the average of every two numbers in the time series. The values in the component "averages" are called wavelet approximation coefficients and the values in the component "differences" are called wavelet detail coefficients. A time series can be reconstructed with the approximation coefficients and detail coefficients. Liu et al. [32] exploits the wavelet analysis and radial basis function neural networks (RBFNN) to extract wave patterns from stock price charts. Wavelet decomposition is applied to both the input data and the template. In their approach, RBFNNs for recognizing patterns at different levels are created. The input of the network is the wavelet-transformed values in a particular resolution. Experiments show that the accuracy of multi-resolution RBFNN matching heavily depends on the resolution level.

3. Classification of chart patterns

In [2], Bulkowski describes the features of 53 chart patterns and their relationship to stock trends. We classify the chart patterns from [2] into five categories in terms of their shapes as shown in Table 4 of Appendix A. The five categories are numbered as C1 to C5. In the following discussion, we outline the important characteristics of chart patterns from different categories and corresponding templates.

Chart patterns can be defined using either points or daily candle sticks. Points (i.e., the closing price of the stock) can be used to represent the first three pattern categories (C1, C2 and C3). C1 and C2 comprise of fluctuation patterns and C3 includes curved patterns. Patterns with fluctuations are further divided into two categories: variable-fluctuation(C1) and fixed-fluctuation (C2) patterns. A daily candlestick is a price stick that consists of the opening, highest and lowest prices in a day. Some daily candlesticks are used to plot the closing price or both the opening and closing prices. Candlesticks are used to represent remaining two categories, spikes (C4) and gaps (C5). In this paper, we do not consider C4 and C5 since the candle stick patterns from these categories require different pattern matching approaches. This is due to the fact that there can be gaps among these candle sticks. We are currently analyzing and testing patterns from C4 and C5 and we are in the process of preparing a separate report on these patterns.

Variable-fluctuation patterns (C1). A variable-fluctuation pattern (C1) has an uncertain number of fluctuations and therefore each pattern in this category has various versions. However, all of these versions should conform to the shape features of the pattern. For example, the "Wedge, Rising" pattern has several variations and the four variants of this pattern are shown in Figures 1(a)-(d). All variations of "Wedge, Rising" have an upward price spiral bounded by two intersecting, up-sloping trend lines. Both trend lines have upward slopes and eventually intersect with the bottom trend line. There are 20 patterns in this group (see Table 4 of Appendix A).

Fixed-fluctuation patterns (C2). A fixed-fluctuation pattern (C2) has a certain number of fluctuations and therefor each pattern in this category can have only one version. For example, as shown in Figure 1(e), a "Head-and-Shoulders Tops" pattern has a three-peak formation with a centre peak taller than the others. The two shoulders appear at about the same price level and the distance from the shoulders to the head is approximately the same. There are ten patterns in this group (see Table 4 of Appendix A).

Curved patterns (C3). For patterns with curves (C3), the shapes form various curves due to the price decline/increase, which results in a gentle rounding turn akin to a half-moon or U/inverted U. For example, Figure 1(f) shows a "Cup with Handle" pattern which is a U-shaped cup with a handle on the right side. There are 16 patterns in this group (see Table 4 of Appendix A).

Candlestick patterns with spikes (C4). Patterns with spikes (C4) has two remarkably longer downward/upward spikes than other downward/upward spikes around them. For each pattern in this category, the highest/lowest prices of two candlesticks are much higher/lower than those of the candlesticks around them. An example of C4 is shown in Figure 1(g). The "Horn Bottom" pattern has two downward spikes separated by a one-week duration in the weekly candlesticks chart. There are four patterns in this group (see Table 4 of Appendix A).

Candlestick patterns with gaps (C5). A gap forms when a candlestick's highest/lowest price is lower/higher than the lowest/highest price of the previous candlestick. An example of patterns with gap (C5) is shown in Figure 1(h). The "Island Long" pattern is formed by several continuous sticks that are separated by the two gaps similar to an island in a long time series represented by candlesticks. There are three patterns in this group (see Table 4 of Appendix A).

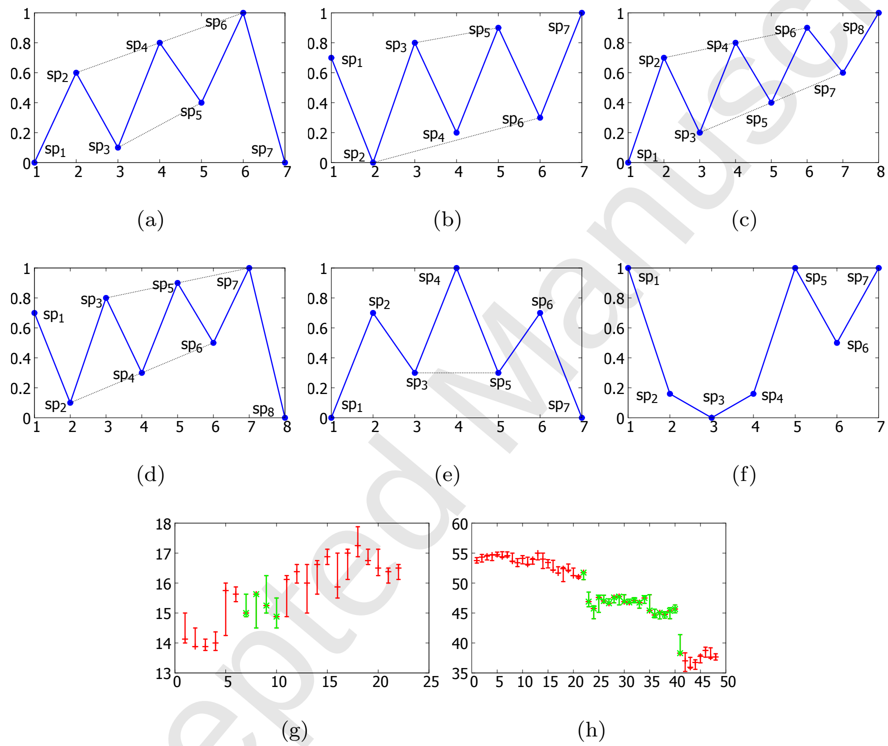


Figure 1: Pattern templates including (a)-(d) "Wedges, Rising"; (e) "Head-and-Shoulders Tops"; (f) "Cup with Handle"; (g) "Horn Bottom" (green); (h) "Island Long" (green).

4. Overview of existing pattern matching approaches

We briefly review the existing pattern matching approaches in this section. In Section 5, we discuss the adaptive neuro fuzzy inference system and how to construct the ANFIS model for chart patterns.

4.1. Template-based pattern matching

The TB pattern matching approach [3] measures the similarity between the defined pattern templates and the sequence in vertical direction and in the horizontal direction by calculating their point-to-point amplitude distance (AD) and temporal distance (TD). AD is defined as

$$AD(S,Q) = \sqrt{\frac{1}{n}\sum_{k=1}^{n}(s_k - q_k)^2} \tag{1}$$

TD is defined as

$$TD(S,Q) = \sqrt{\frac{1}{n-1}\sum_{k=2}^{n}(st_k - qt_k)^2} \tag{2}$$

The similarity measure can be defined as

$$D(S,Q) = w1 \times AD(S,Q) + (1 - w1) \times TD(S,Q) \tag{3}$$

where S denotes the sequence and Q denotes the chart pattern template. $w1$ is the weight required to balance AD and TD. $w1 = 0.5$ for our experiment, which is similar to the weight configured by Fu et al. [3]. s_k and q_k denote the points in S and the pattern template Q, respectively. st_k and qt_k denote the time coordinates of the points s_k and q_k . We consider the sequence S as a pattern when $D(S,Q)$ for a sequence is less than a threshold.

4.2. Rule-based Approach

Predefined rules are used to identify patterns in the RB pattern matching approach [3]. A sequence is recognised as a matching pattern if the sequence conforms to the rules of a given pattern. These rules can be designed according to the descriptions of the shape features in [2]. Adjustments of the rules can be made to tighten or relax the rules for pattern recognition depending on analyst preference.

4.3. Pattern matching based on Euclidean distance

The ED approach measures the similarity between two sequences (the sequence of the template and the segmented sequence) by calculating the point-to-point Euclidean distance between them in vertical direction. The ED between the two sequences $X(x_1, ..., x_n)$ and $Y(y_1, ..., y_n)$ can be represented by the following equation.

$$ED(X,Y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{4}$$

The two sequences are said to be similar if the ED is less than a given threshold.

4.4. Dynamic time warping

The DTW [4] approach measures the similarity between time series that are out of phase. In DTW, given two sequences $X(x_1, ..., x_n)$ and $Y(y_1, ..., y_m)$, an n-by-m matrix M is constructed. The elements $d(x_i, y_j)$ in the matrix represent the Euclidean distance of the points x_i and y_j. $d(x_i, y_j)$ is also called warping cost between two points. The warping path $W = w_1, w_2, ..., w_k, ..., w_K(max(m, n) \leq K<m + n - 1)$ is a neighbouring set of elements in M. The warping path follows three constraints: boundary conditions, continuity and monotonicity. The boundary conditions indicate two conditions, $w_1 = (x_1, y_1)$ and $w_K = (x_n, y_m)$. Continuity means that given $w_k = (a, b)$ and $w_{k-1} = (a', b')$, $a - a' \leq 1$ and $b - b' \leq 1$. Monotonicity indicates that $a - a' \geq 0$ and $b - b' \geq 0$. The optimal warping path $DTW(x, y)$ which minimises the warping cost, is found by dynamic programming. In the dynamic programming, the cumulative distance $\gamma(i, j)$ is defined as the distance $d(x_i, y_j)$, which is found in the current cell and the minimum of the cumulative distances of the adjacent elements.

$$\gamma(i,j) = d(x_i, y_j) + min\{\gamma(i-1, j-1), \gamma(i-1, j), \gamma(i, j-1)\} \tag{5}$$

When $\gamma(n, m)$ is less than a certain threshold, the two sequences are said to be similar.

4.5. Pattern matching with a support vector machine

A synthetic dataset with different class labels is used for training a SVM model. Using the trained SVM model, a sequence can be classified into one of the labeled classes. LIBSVM [24] which is a popular open-source machine learning library for the SVM is used in our experiment. We input a set of training data (i.e., the time series of different patterns) and use the C-Support

Vector Classification (C-SVC) in the LIBSVM to build a model to classify the given pattern. Each time series is a vector. The positive time series are labelled as 1 and the negative time series are labelled as -1. Suppose there is a training vector $x_i \in R^n, i = 1, ..., l$, and an indicator vector $y \in R^l$ such that $y_i \in \{-1, 1\}$, C-SVC solves the following optimization problem:

$$\min_{w,b,\xi} \frac{1}{2} w^T w + C \sum_{i=1}^{l} \xi_i \tag{6}$$

subject to $y_i(w^T \phi(x_i) + b) \geq 1 - \xi_i$, $\xi_i \geq 0$, $i = 1, ..., l$.

where $\phi(x_i)$ maps x_i into a higher dimensional space and C comprises the regularisation parameters. w is the vector variable which is going to be determined after the training and w^T is a transposed matrix of w.

4.6. Pattern matching with Hidden semi-Markov model

Ge et al. [5] detect specific patterns of shapes in time series data with Hidden semi-Markov modle. The pattern of interest can be modeled as a K-state segmental hidden Markov model. Each state is responsible for the generation of a component of the overall shape. A recursive Viterbi-like algorithm is used to find the optimal state sequence of a given time series.

Based on the idea of [5], we design Hidden-semi Markov models (HSMM) by using the template of each chart pattern. Each line segment of the template is a state of the model. The line segment is modeled by a linear regression function of time t added with noise e. y in equation 7 is the observation of duration d (i.e., the number of points) generated by a state k.

$$y = a_k + b_k t + e_k \tag{7}$$

where a_k is the intercept, b_k is the slope of the linear regression function and e_k is the mean square error in the k state. The output probability of an observation y_i in state k, denoted by $c_k(y_i) = p(y_i \mid \mu_k, \sigma_k^2)$, is calculated by the probability density function of state k. μ_k is the linear regression function of time t in state k. In other words, $\mu_k = a_k + b_k t$. σ_k^2 is defined as the mean square error (MSE) of $\hat{y}'$ (the value of the linear regression function) as fitted to the original data y, i.e.,$\sigma_k^2 = \frac{(y-y')^2}{d}$. There are d Gaussian distributions with different μ and the same σ^2 in a state. Note that the time t is a variable although the intercept and the slope are fixed in a state.

The duration of a state is indicated by the number of data points in that state. Instead of modelling the number of points in a state, we model the length of a state (i.e., the length of a line segment in the state) using a Poisson distribution:

$$p(l \mid \lambda_k) = \begin{cases} 1, & d = 1 \\ \frac{\lambda_k^l}{l!} exp(-\lambda_k), & d > 1 \end{cases} \tag{8}$$

where l is the length of a sequence of observations calculated by the ED of the start and end points of the sequence of observations and λ_k is the length of a state (i.e., the length of the line segment of state k). Any two line segments of the same length may have different numbers of points and may also have the same $p(l \mid \lambda_k)$.

The probability of a sequence of observations of d points generated by state k is calculated as

$$p(y \mid \mu_k, \sigma_k^2, \lambda_k) = p(l \mid \lambda_k) \prod_{i=1}^{i=d} p(y_i \mid \mu_k, \sigma_k^2) \tag{9}$$

The Viterbi algorithm [8] can be used to find the optimal state sequence associated with the given observation sequence. The optimal state sequence is the state sequence that maximises $p(y \mid \mu_k, \sigma_k^2, \lambda_k)$ (given in equation 9). To find a single best state sequence $S = (s_1, s_2, ..., s_T)$ for the given observation sequence $y = (y_1, y_2, ..., y_T)$, we first define a quantity $\delta_t(i)$ to explain the Viterbi algorithm.

At time t in state i, $\delta_t(i) = \underset{s_1, s_2, ... s_{t-1}}{max} P[s_1 s_2 ... s_t = i, y_1 y_2 ... yt \mid \mu_k, \sigma_k^2, \lambda_k]$ denotes the maximum probability to generate a sequence of observations $(y_1, y_2, ... y_t)$ along a single state path $(s_1, s_2, ... s_{t-1})$. By induction, we can calculate the following:

$$\delta_t(i) = \underset{d}{max}[\underset{j}{max} \delta_{t-d}(j) a_{ji} p(l \mid \lambda_i, d) \prod_{s=t-d+1}^{s=t} c_i(y_s)] \tag{10}$$

If the final state of the optimal state sequence of the given observation sequence is the same as the predefined final state, the observation sequence is considered as a pattern.

5. Adaptive neuro fuzzy inference system (ANFIS)

5.1. The architecture of ANFIS

A fuzzy if-then rule consists of the linguistic variables and linguistic values which are characterized by membership functions. The form of a fuzzy if-then rule is *"if x is V_1, then y is V_2"*, where x and y are linguistic variables and V_1 and V_2 are linguistic values characterized by membership functions. The "*if*" part of the rule is called the premise part and the "*then*" part of the rule is called the consequent part. Fuzzy reasoning is an inference procedure used to derive conclusions from a set of fuzzy rules. In Takagi and Sugeno's fuzzy reasoning model [33], the output of each rule is a linear combination of input variables plus a constant value, and the final output is the weight average of the outputs of each rule.

Figure 2(a) shows an example of the Takagi and Sugeno's fuzzy reasoning for two rules with two inputs (x_1 and x_2) and Figure 2(b) is the equivalent architecture of adaptive neuro fuzzy inference system (ANFIS). According to Figure 2(a),

- Rule 1 denotes "if x_1 is A_1 and x_2 is B_1, then the output is $f_1 = p_1x_1 + q_1x_2 + r_1$."
- Rule 2 denotes "if x_1 is A_2 and x_2 is B_2, then the output is $f_2 = p_2x_1 + q_2x_2 + r_2$."

The premise parts (the "if" parts) of the two rules correspond to the Layer 1 in Figure 2(b). In Layer 1, all the nodes are rectangular nodes indicating adaptive nodes, the parameters of which need to be identified with learning algorithm. A_i and B_i are linguistic values that are characterized by membership functions. The Gaussian membership function [10] is given by:

$$\mu_{A_i} = e^{-(\frac{x-c_i}{a_i})^2} \tag{11}$$

where c_i and a_i are called premised parameters. The outputs of the Layer 1 are the membership grades ($\mu_{A_1}(x_1)$, $\mu_{B_1}(x_2)$, $\mu_{A_2}(x_1)$ and $\mu_{B_2}(x_2)$) of the inputs. Therefore,

- Rule 1 can be represented as "if x_1 is $\mu_{A_2}(x_1)$ and x_2 is $\mu_{B_1}(x_2)$, then the output is $f_1 = p_1x_1 + q_1x_2 + r_1$."
- Rule 2 can be represented as "if x_1 is $\mu_{A_2}(x_1)$ and x_2 is $\mu_{B_2}(x_2)$, then the output is $f_2 = p_2x_1 + q_2x_2 + r_2$."

All nodes in Layer 2 are circle nodes indicating fixed nodes. In Layer 2, the outputs of Layer 1 are multiplied to produce w_i which is given by:

$$w_i = \mu_{A_i}(x_1) \times \mu_{B_i}(x_2), i = 1, 2 \tag{12}$$

The normalization of the outputs of Layer 2 is implemented in Layer 3. In Layer 3,

$$\bar{w_i} = \frac{w_i}{w_1 + w2}, i = 1, 2 \tag{13}$$

The consequent parts (the "*then*" parts) of the two rules correspond to Layer 4. In this layer, all nodes are adaptive nodes and the parameters of these nodes need to be identified with the learning algorithm. The output of Layer 4 is given by:

$$\bar{w}_i f_i = \bar{w}_i(p_i x_1 + q_i x_2 + r_i), i = 1, 2 \tag{14}$$

where p_i, q_i and r_i are called consequent parameters.

The final output is calculated in Layer 5. In this layer, all the outputs in Layer 4 are summed as follows:

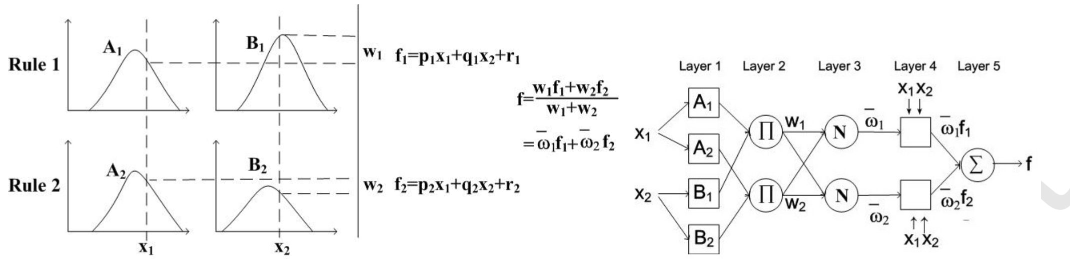
$$f = \sum_i \bar{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \tag{15}$$

A hybrid learning approach [9] which combines the gradient method and the least squares estimate is used to identify the premise parameters (c_i and a_i in equation 11) and the consequent parameters (p_i, q_i and r_i in equation 14). At each epoch, the update of parameters takes place after the whole training data set is presented. The identification process is composed of a forward pass and a backward pass at each epoch. In the forward pass, the premise parameters are fixed and the input data are presented to the system to calculate the actual output. The least squares estimate is used to find the optimal consequent parameters to minimize the squared error of the actual output and the expected output. The backward pass begins after the optimal consequent parameters are identified. In the backward pass, the error measures propagate from the output end to the input end. The premise parameters are updated by the gradient method.

Before the training process of an ANFIS, an initial fuzzy inference system is provided by the subtractive clustering technique ([11], [34]) which is used to locate the cluster centers of the input output data pairs. The number of clusters determined by subtractive clustering algorithm is the number of rules in fuzzy inference system. The subtractive clustering technique determines the number of rules and initial values of the premise parameters. The least squares estimation is used to determine each rule's consequent parameters. The initial fuzzy inference system contains a set of fuzzy rules to cover the feature space.

5.2. The training and testing process of ANFIS model for chart patterns

We generate deformations of three chart patterns ("Wedges, Rising", "Head-and-Shoulders Tops" and "Cup with Handle") from the three categories (C1, C2 and C3) to be used as synthetic datasets. Templates in Figure 1(a), (e) and (f) are represented by a several important



(a) (b)

Figure 2: (a) Takagi and Sugeon's fuzzy reasoning [10]. (b) The corresponding architecture of ANFIS [10].

points. The generation of synthetic datasets is based on these important points from the templates. In addition, these templates are also used as the standard templates for the TB approach. Time scaling, time warping and noise adding steps are used in the syntectic data generation algorithm described in [3] and [35]. Since we regard the important points of the template as the features of a chart pattern, it is only necessary to generate the deformations of the template to obtain synthetic datasets. Therefore, we generate deformations that have the same length as the template. In this paper, we use the "noise adding" ([3], [35]) to change the positions of important points to make more deformations of the template. In the generated synthetic datasets, the deformations of the templates are regarded as positive cases and each positive case is normalized.

In Figure 3, we depict the training and testing process of the ANFIS model. Detailed discussions about the whole training and testing process are given in following sections.

5.2.1. The training process

After the generation of synthetic datasets, the ANFIS model is trained and thresholds for different chart patterns are determined. In the first step, we train the ANFIS model with the training dataset. The expected outputs or the labels for the three chart patterns are set to 1, 2 and 3 respectively. The training dataset is made up of 3000 cases (1000 positive cases of "Wedges, Rising" with label 1, 1000 positive cases of "Head-and-Shoulders Tops" with label 2 and 1000 positive cases of "Cup with Handle" with label 3). Since the three chart patterns have 7 important points respectively, the training dataset has 8 columns. The first seven columns are the important points and the last column is the expected outputs or labels (1 for "Wedge, Rising", 2 for "Head-and-Shoulders Tops", and 3 for "Cup with Handle"). The MATLAB Fuzzy Logic Toolbox is used

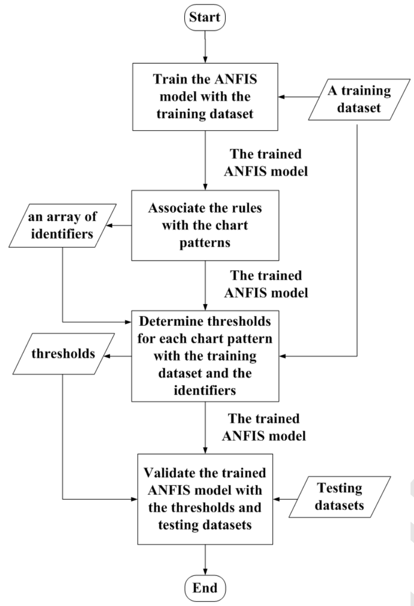


Figure 3: The training and testing process of the ANFIS model.

to train the ANFIS model. Subtractive clustering is adopted to generate the initial fuzzy inference system and the radius is set to 0.2. Finally, we obtain the trained ANFIS model. In addition, the premise and consequent parameters are identified. Algorithm 1 describes the training process with the MATLAB Fuzzy Logic Toolbox. In Algorithm 1, the initial fuzzy inference system is generated with the function "genfis2" (line 3). Then the function "anfis" is used to tune the parameters based on the initial fuzzy inference system with the training dataset (line 4). "genfis2" and "anifs" in Algorithm 1 are two functions from the MATLAB Fuzzy Logic Toolbox [36]. "genfis2" generates a Sugeno-type fuzzy inference system structure using subtractive clustering. "anfis" uses a hybrid learning algorithm to tune the parameters of a Sugeno-type fuzzy inference system.

5.2.2. Association of the rules from the trained ANFIS model with the chart patterns

In the second step, we associate the rules from the trained ANFIS model with the chart patterns. The subtractive clustering technique determines the number of clusters or the number of rules of the trained ANFIS model. After the training process, the trained ANFIS model has 13 rules. In other words, the training dataset is divided into 13 clusters with the subtractive clustering technique. Each rule has seven inputs and each input is described by one gaussian membership

Algorithm 1 Train ANFIS

```
Function: TrainANFIS(trnData, radii)
Input: trnData: the training dataset containing n columns
    radii: radius of the subtractive clustering centers
Output: fis: the trained ANFIS model
Xin ← the first n − 1 columns of the trnData, Xout ← the last column of the trnData
iniFIS ← genfis2(Xin, Xout, radii)
fis ← anfis(trnData, initFIS)
return fis
```

function. These rules are used to describe different chart patterns. A particular chart pattern can be described by more than one rule. To associate a rule with a particular chart pattern, the seven means (c_i in equation 11) of the gaussian membership functions of a rule are presented to the trained ANFIS model. Then, we get an output of the ANFIS model. If the rounded output is 1, then the rule is used to describe the "Wedger, Rising". If the rounded output is 2, then the rule is used to describe the "Head-and-Shoulders Tops". If the rounded output is 3, then the rule is used to describe the "Cup with Handle". The process of associating rules to a particular chart pattern is shown in Algorithm 2. For each rule in the trained ANFIS model, the means of the gaussian membership functions of each input are retrieved (line 4). Then the retrieved means are presented to calculate an output (f in equation 15) with the function "evalfis" from the MATLAB Fuzzy Logic Toolbox (line 5) which performs fuzzy inference calculations. Next, the output is rounded to compare with the particular label (line 6 and line 7). This process is repeated until all the rules in the ANFIS model are checked. Finally, we obtain the array of identifiers of rules for each chart pattern. We found that the first, fourth, sixth, eleventh and thirteenth rules are used to describe "Wedge, Risings" pattern. The second, fifth, seventh and ninth rules are used to describe "Head-and-Shoulders Tops" pattern. The third, eighth and twelfth rules are used to describe "Cup with Handle" pattern.

5.2.3. Calculation of thresholds

In the third step, we calculate an array of thresholds for each chart pattern. Thresholds are used to represent the minimum degrees of satisfying the rules describing a particular chart pattern. In section 5.2.2, we associate each rule with a particular chart pattern. As shown in equation 15,

Algorithm 2 Associate rules

Function: AssociateRules(fis, label)

Input: fis: the trained ANFIS model

label: the label of a particular chart pattern

Output: identifiers_array: an array of identifiers of the rules used to describe a particular chart pattern

$m \leftarrow$ the number of rules in "fis"

for i=1 to m **do**

  input_means $\leftarrow$ the means of the gaussian membership functions of the rule i

  output $\leftarrow$ evalfis(input_means, fis)

  rounded_output $\leftarrow$ round the "output"

  if rounded_output == label **then**

    add the identifier i to identifiers_array

  end if

end for

return identifiers_array

the output of each rule f_i is weighted by w_i called "aggregation value". The calculation of w_i is shown in equation 12 in which the membership grade of each input is used in multiplication. For an input sequence, if the aggregation value w_i of a rule i is greater than the values of other rules, then the input sequence is likely to be the chart pattern which is described by the rule i. Therefore, we calculate the minimum value of w_i for each rule i using the training dataset. Next, the calculated minimum value of w_i is used as a threshold to determine the class of chart pattern for the testing dataset.

The 3000 positive cases from the training dataset are divided into three sub-datasets and presented to the trained ANFIS model separately without the labels. Algorithm 3 shows the process of determining the thresholds for a particular chart pattern. A sub-dataset containing the cases of a particular chart pattern is obtained from the training dataset (line 2). Each case in the sub-dataset is presented to calculate the aggregation values of each rule (w_i in equation 12) of the trained ANFIS model (line 6 and line 7). Then we get the aggregation array "arr" which contains 13 aggregation values. The aggregation values of the rules which are used to describe a particular chart pattern are retrieved according to "identifiers_array" of the Algorithm 2 (line 9 to line 11). The array "tempArr" is used to store the minimum aggregation values of the rules which are used to describe a particular chart pattern (line 12 to line 16). For example, if we want to determine the thresholds of "Wedges, Rising", first the 1000 positive case of "Wedges, Rising" without labels are presented to the trained model. For each input positive case, the aggregation value of each rule (w_i in equation 12) is calculated. After the 1000 positive case presented to the ANFIS model, there is 1000 aggregation values for each of the 13 rules. At the second step, since we have identified that "Wedge, Rising" is described by five rules (the first, fourth, sixth, eleventh and thirteenth rules), we can retrieve the minimum values from the 1000 aggregation values of the five rules respectively. Then we obtain five minimum aggregation values. These five minimum aggregation values are the thresholds for "Wedge, Rising". These thresholds represent the minimum degrees of satisfying the rules describing "Wedge, Rising" pattern.

5.2.4. The testing process

In the last step, we use test datasets to validate the ANFIS model. We generated another 5000 positive cases without labels for each chart pattern. In addition, a testing dataset consist of 5000 negative cases is also generated to test the trained ANFIS model. Each negative case is a

Algorithm 3 Calculate an array of thresholds for a particular chart pattern

```
Function: GetThresholds(trnData, fis, identifiers_array, label)
Input: trnData: the training dataset
    fis: the trained ANFIS model
    identifiers_array: an array of identifiers of the rules used to describe a particular chart patter
    label: the label of a particular chart pattern
Output: thresholds: an array of thresholds for a particular chart pattern
tempData ← the trnData's subset containing the cases with the specified "label"
m ← the length of "identifiers_array", n ← the number of cases of "tempData"
tempArr ← an array containing m infinite numbers
for i=1:n do
    tempInput ← the case i of "tempData"
    arr ← evalfis(tempInput, fis) \\ "arr" is an array of aggregations
    sub_arr ← an empty array of length m
    for k=1:m do
        sub_arr[k] ← arr[identifiers_array[k]]
    end for
    for j=1:m do
        if sub_arr[j]≤tempArr[j] then
            tempArr[j] ← sub_arr[j]
        end if
    end for
end for
thresholds ← tempArr
return thresholds
```

sequence of 7 data points randomly generated between 0 and 1. In total, three testing datasets are prepared for the three chart patterns. Each testing dataset has 5000 positive cases of a chart pattern and 5000 negative cases. Algorithm 4 shows the testing process for a particular chart pattern. For each case in the testing dataset, the output (equation 15) and the aggregation values (equation 12) are calculated with the function "evalfis" (line 4 and line 5). Next, the maximum value of the 13 aggregation values in the array "arr" and the index of the maximum value in the array "arr" are retrieved (line 6). Then we check if the index of the maximum value is one of the elements of the array "indentifiers_array" (line 7). We compare the rounded output with the label. Finally, the maximum value is compared with the threshold (line 9). For example, to test whether a given sequence is a pattern of "Wedge, Rising" or not, the sequence is first input to the trained ANFIS model. Next, we obtain one output (equation 15) of the ANFIS model and 13 values of aggregation (equation 12) for the 13 rules. After that, the algorithm checks whether the rounded output (equation 15) of the model is equal to 1 or not. Next, the algorithm compares whether the index of the maximum value of aggregations of the 13 rules is one of the five rules used to describe "Wedge, Rising" as shown in the second step. Finally, the algorithm checks whether the maximum value of aggregation is greater than the corresponding threshold. If the new input satisfy the three conditions, then the new input is classified as a "Wedge, Rising". By using above three conditions, the ANFIS model can better exclude the negative cases and improve its classification accuracy.

6. Experiments

The experiments are performed on synthetic datasets and a real datadataset. The algorithms of SVM, TB, RB, ED, DTW and HSMM are coded in JAVA programming language on a computer with Intel(R) Core(TM) i7-4790 CPU 3.60GHz processor, 8 GB RAM, and 64-bit Microsoft Windows 7 operation system. The ANFIS method is coded with Matlab Fuzzy Logic Toolbox on the same computer.

For experiments on synthetic datasets, we generate the synthetic datasets for the three chart patterns from C1, C2 and C3. The three chart patterns are: "Wedges, Rising" in Figure 1(a), "Head-and-Shoulders Tops" in Figure 1(e) and "Cup with Handle" in Figure 1(f). For C1, we choose one of the variations (Figure 1(a)) to evaluate the different pattern matching approaches. As stated in section 3, we do not consider the patterns from C4 and C5 for our experiments.

Algorithm 4 Testing the ANFIS model

```
Function: TestANFIS(testData, fis, label, identifiers_array, thresholds)
Input: testData: testing dataset (the previous n cases are positive cases and the remaining are
    negative cases)
    fis: the trained ANFIS model
    label: the label of a particular chart pattern
    identifiers_array: an array of identifiers of the rules used to describe a particular chart pattern
    thresholds: an array of thresholds for a particular chart pattern
Output: accuracy: the accuracy of testing the ANFIS model for a particular chart pattern
m ← the number of cases in "testData"
for i=1 to m do
  tempCase ← the case i in "testData"
  [output, arr] ← evalfis(tempCase, fis)
  max_value ← maximum value of the array "arr", max_index ← the index of the maximum
  value of the array "arr"
  if "max_index" is one of the elements of "identifiers_array" then
    j ← the index of the elements containing "max_index" in the array "idenfiers_array"
    if the rounded output == label && max_value ≥ thresholds[j] then
      tempCase is recognized as a chart pattern
    end if
  end if
end for
return accuracy
```

For experiments on the real dataset, we locate and classify the three chart patterns ("Wedge, Rising", "Head-and-Shoulders Tops" and "Cup with Handle") using different pattern matching approaches. To locate a chart pattern in a real financial time series, a sliding window shifts along the time series to extract a subsequence from the time series. Then the subsequence is segmented to the same length as of the template. Finally, pattern matching approaches are used to classify the segmented sequence.

In the experiments, the perceptually important points (PIP) method [18] is used as the segmentation method to preprocess the sequence. With the time series T, the first and the last data point in the time series are the first two PIPs. The third PIP is the point in T with maximum vertical distance to the line joining the first two PIPs. The fourth PIP is the point in T with maximum vertical distance to the line joining its two adjacent PIPs, either between the first and second PIPs or between the second and the last PIPs. This process continues until the length of the segmentation sequence is equal to the template.

6.1. Experiments on synthetic datasets

The training and testing process of the ANFIS model are shown in subsection 5.2. To compare with the SVM method, we train a SVM model with the training dataset of 4000 cases(1000 positive cases of "Wedges, Rising", 1000 positive cases of "Head-and-Shoulders Tops", 1000 positive cases of "Cup with Handle" and 1000 negative cases). The training dataset consists of 3000 positive cases of the three chart pattern which are the same as the 3000 cases in the training dataset of the ANFIS model and 1000 negative cases of points randomly generated between 0 and 1. When a new input is presented to the SVM, the new input will be assigned to one of the classes of the training dataset. Therefore if we want to test the datasets of negative cases, we need to train the SVM with the datasets of negative cases beforehand. While for the ANFIS model, only positive cases are needed in the training datasets.

The thresholds of TB for "Wedge, Rising", "Head-and-Shoulders Tops" and "Cup with Handle" are 0.1, 0.1 and 0.16 respectively. The thresholds of ED for "Wedge, Rising", "Head-and-Shoulders Tops" and "Cup with Handle" are 0.3, 0.3 and 0.5 respectively. The thresholds of DTW for "Wedge, Rising", "Head-and-Shoulders Tops" and "Cup with Handle" are 0.3, 0.3 and 0.3 respectively. We determine the thresholds for TB, ED and DTW with the same training datasets as the SVM model's. Figure 4 shows an example of calculating the threshold of ED for the chart pattern

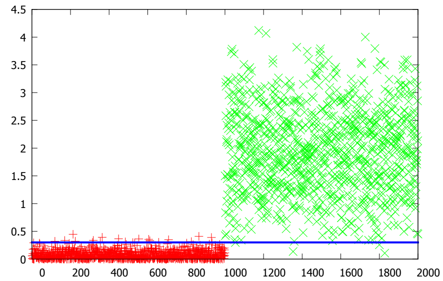


Figure 4: An example of determining the threshold of ED for the chart pattern “Cup with Handle”. The Y-axis denotes the distance and the X-axis denotes the identifier of the case in the training dataset.

“Cup with Handle”. The distances between the cases in the training dataset and the template are calculated with ED. The red points are the distances between the “Cup with Handle” template and the positive cases calculated with the approach ED. The green points are the distances between the “Cup with Handle” template and the negative cases calculated with the approach ED. The blue line is the threshold that separates two kind of distances. The threshold of ED is 0.3 for the chart pattern “Cup with Handle”. If the value of the distance between the “Cup with Handle” template and a new input calculated by ED is less than 0.3, the new input is identified as a pattern of “Cup with Handle”. The other thresholds can be determined in a similar way.

The testing datasets for SVM, TB, RB, ED, DTW and HSMM are the same as the three testing cases for ANFIS. Each testing dataset has 5000 positive cases and 5000 negative cases. ANFIS takes 4681 milliseconds to train while SVM takes 272 milliseconds to train. ANFIS is slower than the SVM model in training. Table 1 shows the time used to test the three testing cases with seven pattern matching approaches. ANFIS model also takes more time for testing compared to other approaches. Each approach takes similar time to test the three different testing datasets. The slower training time for ANFIS model is possibly caused by the application of gradient descent in the hybrid learning algorithm to identify the parameters. TB, RB ED and DTW is more efficient than ANFIS, SVM and HSMM in Table 1. RB is efficient because it just needs to check the predefined rules which define the relative positions of the points in the sequence. For TB, ED or DTW, the distance calculation results are compared with the predefined thresholds. Therefore, these approaches are faster than ANFIS, SVM and HSMM. ANFIS checks more conditions to classify a chart pattern, thus it takes more time to test than other pattern matching approaches.

Figure 5 shows the accuracy, recall and precision of the seven pattern matching approaches for

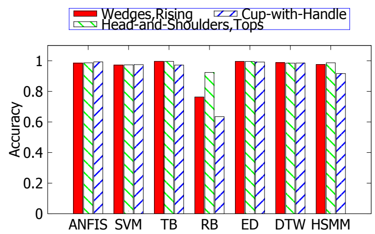
Table 1: The testing time of different pattern matching methods for the three chart patterns. The time unit is in millisecond(ms).

	Wedges, Rising	Head-and-Shoulders, Tops	Cup with Handle
ANFIS	7549	7514	7499
SVM	627	640	627
TB	158	152	133
RB	59	69	69
ED	31	31	31
DTW	47	62	47
HSMM	755	830	789

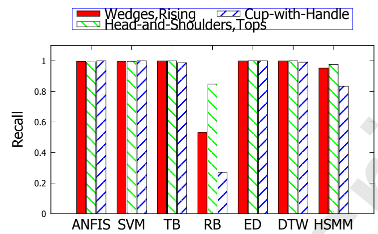
different chart patterns. In Figure 5(a), the six approaches have similar high accuracy except the RB approach. The RB approach has a high precision (see Figure 5(c)) but a low recall (see Figure 5(b)). The RB approach is a rigid approach and likely to accept more true negatives as well as false negatives than other approaches. Recall that SVM and ANFIS have been trained with the training datasets beforehand. TB, ED, DTW determine the thresholds with the training datasets. Therefore their accuracy, recall and precision are at a similar level.

6.2. Experiments on the real dataset

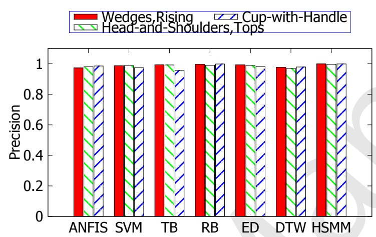
In this experiment, we classify "Wedges, Rising", "Head-and-Shoulders Tops" and "Cup with Handle" patterns in the historical daily closing prices of the Hang Seng Index (HSI) from 3 January 2005 to 31 December 2015, which contains 2,759 points [37]; the NYSE AMEX Composite Index (NYSE) from 1 February 2005 to 31 December 2015, which contained 2,769 points; and the Dow Jones Industrial Average (DJI) from 1 February 2005 to 31 December 2015, which contained 2,769 points. A sliding window shifts one data point at a time to obtain a sub-sequence from HSI. For example, in Figure 6(a), a sliding window of length 30 shifts one data point at a time to get a sub-sequence from the time series of the historical prices. The sliding window extracts a length of 30 points from the time series as shown in Figure 6(b). Then PIP segmentation method is performed to extract seven points among these 30 points in Figure 6(c). Finally, the normalized sequence of seven points is input to different pattern matching methods. Redundant patterns that are within +1 or -1 difference in position are eliminated.



(a)

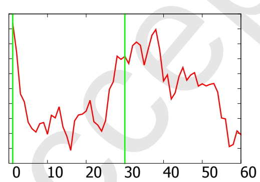


(b)

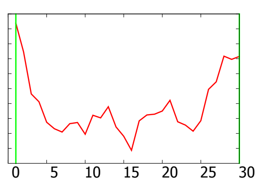


(c)

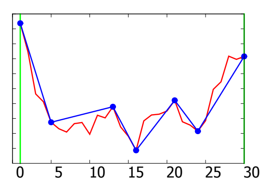
Figure 5: (a), (b) and (c) are the accuracy, recall and precision of different pattern matching methods for the testing datasets.



(a)



(b)



(c)

Figure 6: (a) A time series of length 60 (red). The two vertical green lines form a sliding window of length 30. The sliding window shifts one data point at a time. (b) The green sliding window extracts the subsequence (the red line) from the original time series. (c) The segmentation result (blue) of the subsequence.

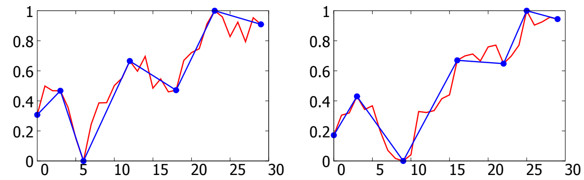
Table 2: The number of patterns found by seven pattern matching approaches in the historical price of Hang Seng Index (HSI), NYSE AMEX Composite Index (NYSE) and Dow Jones Industrial Average (DJI). We use a triplet [α,β,γ] to express the number of patterns found in these three stock indexes. α, β and γ are the number of patterns found in HSI, NYSE and DJI respectively.

	Wedges, Rising	Head-and-Shoulders Tops	Cup with Handle
ANFIS	[28,29,38]	[23,27,16]	[11,10,9]
SVM	[7,7,11]	[17,21,13]	[22,24,21]
TB	[2,0,0]	[5,3,0]	[7,11,6]
RB	[12,20,19]	[9,10,13]	[2,0,1]
ED	[11,9,14]	[22,25,16]	[15,15,15]
DTW	[16,16,19]	[32,32,27]	[14,23,23]
HSMM	[3,2,1]	[6,5,3]	[2,0,0]

Table 2 shows the number of patterns found by the seven pattern matching approaches in HSI, NYSE and DJI. In this paper, we discuss the patterns found in HSI. For the patterns found in HSI, ANFIS found the highest number of “Wedge, Rising” patterns. DTW found the highest number of “Head-and-Shoulders Tops” patterns. All the “Wedge, Rising” and “Head-and-Shoulders Tops” patterns found by SVM, TB and ED are also found by DTW. SVM found the highest number of “Cup with Handle” patterns. All the “Cup with Handle” patterns found by ANFIS are also found by SVM. TB found the least number of “Wedge, Rising” and “Head-and-Shoulders Tops” patterns. All the “Wedge, Rising” and “Head-and-Shoulders Tops” patterns found by TB are also found by ANFIS, SVM, ED and DTW. RB and HSMM found the least number of “Cup with Handle” patterns. All the “Cup with Handle” patterns found by RB and HSMM are also found by the other five approaches. In the following figures, all the subsequences found are normalized. To further analyse the experiment results from Table 2, the detailed characteristics of the patterns found in HSI by each pattern matching approach are elaborated in the following sections.

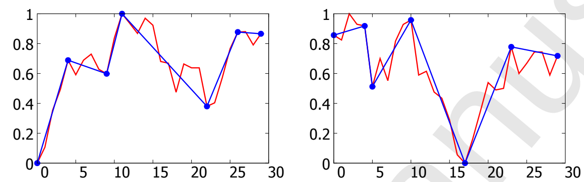
6.2.1. Patterns found by RB

From the experiment results given in Table 2, seven out of 12 “Wedge, Rising” patterns found by RB are not found by any other five approaches. Two out of the nine “Head-and-Shoulders Tops” patterns found by RB are also not found by other five approaches. RB approach defines the relative amplitude positions of the points of the “Wedge, Rising” patter. However, in our rules of “Wedge,



(a) (b)

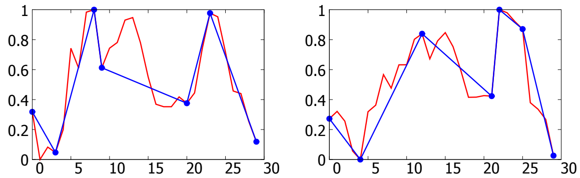
Figure 7: (a) The segmentation result (blue) on a normalized subsequence (red) from 26 September 2011 to 7 November 2011 in HSI found by RB as a "Wedge, Rising" pattern. (b) The segmentation result (blue) on a normalized subsequence (red) from 5 July 2006 to 15 August 2006 in HSI found by RB as a "Wedge, Rising" pattern.



(a) (b)

Figure 8: (a) The segmentation result (blue) on a normalized subsequence (red) from 2 September 2009 to 13 Octorber 2009 in HSI found by RB as a "Head-and-Shoulders Tops" pattern. (b) The segmentation result (blue) on a normalized subsequence (red) from 24 November 2014 to 7 January 2015 in HSI found by RB as a "Head-and-Shoulders Tops" pattern.

Rising", there are no rules about the temporal positions of the points. That means there is no constraint on the temporal positions of the points. Therefore, the "Wedge, Rising" patterns found only by RB (see Figure 7(a) and (b)) have three peaks of various widths when compared with the template from Figure 1(a). If we define rules about the temporal positions of the points to make the peaks proportional, the RB approach becomes too rigid to find any "Wedge, Rising" pattern. Likewise, the RB approach defines rules about the amplitude positions and temporal positions of points of the pattern "Head-and-Shoulders Tops". The relative positions of the third (sp_3) and fifth (sp_5) points as shown in the template of "Head-and-Shoulders Tops" pattern (see Figure 1(e) are not restricted in our rules. Therefore in Figure 8(a) and (b), the third and fifth points are not at the same price level. Although more rules can be added to align the patterns found by RB with the predefined template in Figure 1(e), such enforcement could make the RB more rigid in classification.



(a) (b)

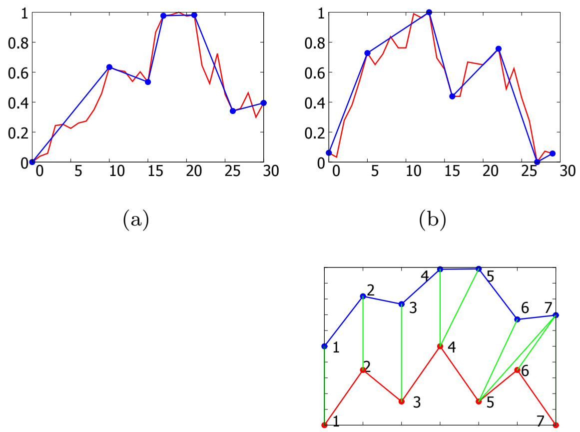
Figure 9: (a) The segmentation result (blue) on a normalized subsequence (red) from 1 December 2008 to 13 January 2008 in HSI found by DTW as a “Wedge, Rising” pattern. (b) The segmentation result (blue) on a normalized subsequence (red) from 4 August 2014 to 15 September 2014 in HSI found by DTW as a “Wedge, Rising” pattern.

6.2.2. Patterns found by DTW

From the experiment results given in Table 2, four out of 16 “Wedge, Rising” patterns found by DTW are not found by any other five approaches. Seven out of 32 “Head-and-Shoulders Tops” patterns found by DTW are not found by any other five approaches. Eight out of 14 “Cup with Handle” patterns found by DTW are not found by the other five approaches. “Wedge, Rising” (Figure 9(a) and (b)), “Head-and-Shoulders Tops” (Figure 10(a) and (b)) and “Cup with Handle” (Figure 11(a) and (b)) patterns which are only found by DTW show high deformation when compared to the templates shown in Figure 1(a), (e) and (f). One of the reasons is that DTW maps a point from the template to more than one point of the query sequence or vice versa. Figure 10(c) depicts the template of “Head-and-Shoulders Tops” and the segmentation result in shown Figure 10(a). Figure 10(d) shows that the fourth point of the template maps to the fourth and the fifth points of the segmentation result. The fifth and sixth points of the template also map to the seventh point of the segmentation result. DTW maps one point of the template to more than one point in the segmented sequence, therefore DTW is able to accept more deformations of the pattern. However, some patterns found are dissimilar to the shape of the templates. As shown in Figure 9(a) and (b), Figure 10(a) and (b), Figure 11(a) and (b), “Wedge, Rising”, “Head-and-Shoulders Tops” and “Cup with Handle” patterns found only by DTW by no means conform to the shapes of “Wedge, Rising”, “Head-and-Shoulders Tops” and “Cup with Handle”.

6.2.3. Patterns found by ED

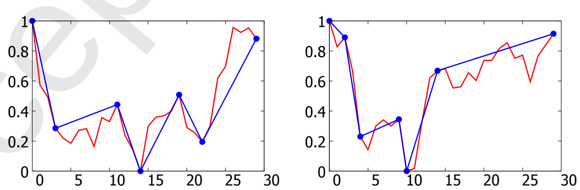
From the experiment results given in Table 2, one out of 15 “Cup with Handle” patterns found by ED are not found by the other pattern matching approaches. ED calculates the point-to-point



(a) (b) (c)

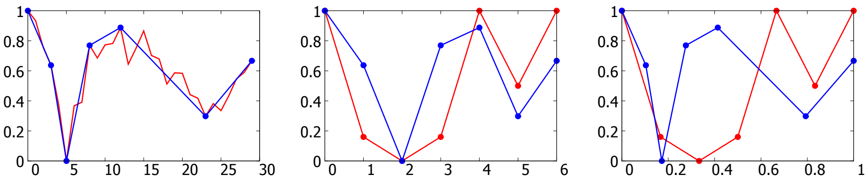
(d)

Figure 10: (a) The segmentation result (blue) on a normalized subsequence (red) from 19 July 2005 to 29 August 2005 in HSI found by DTW as a “Head-and-Shoulders Tops” pattern. (b) The segmentation result (blue) on a normalized subsequence (red) from 17 April 2008 to 28 May 2018 in HSI found by DTW as a “Head-and-Shoulders Tops” pattern. (c) The segmentation result (blue) from (a) and the template of the “Head-and-Shoulders Tops” pattern (red). (d) The mappings of points between the segmentation result (blue) from (a) and the template (red) with the DTW approach.



(a) (b)

Figure 11: (a) The segmentation result (blue) on a normalized subsequence (red) from 4 January 2005 to 17 February 2005 in HSI found by DTW as a “Cup with Handle” pattern. (b) The segmentation result (blue) on a normalized subsequence (red) from 10 May 2005 to 21 June 2005 in HSI found by DTW as a “Cup with Handle” pattern.



(a) (b) (c)

Figure 12: (a) The segmentation result (blue) on a normalized subsequence (red) from 20 October 2008 to 28 November 2008 in HSI found by ED as a "Cup with Handle" pattern. (b) The segmentation result (blue) and the template of the "Cup with Handle" pattern (red). (c) The segmentation result (blue) and the template of the "Cup with Handle" pattern (red). In sub-figure (c), both of the segmentation result and the template are normalized to the range between 0 and 1 in the direction of X-axis.

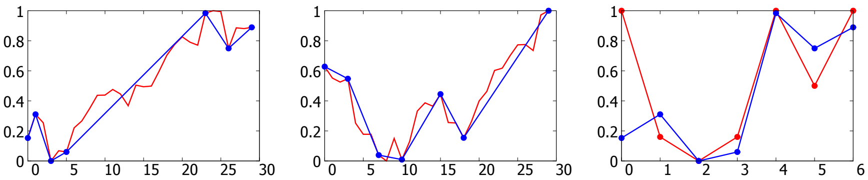
distance between the template and the segmentation result in the vertical direction. Figure 12(a) shows a "Cup with Handle" pattern only found by ED. In Figure 12(b), five points (among seven points) from the segmentation result (blue) are similar to the corresponding points from the "Cup with Handle" template (red) in vertical direction except the second and the fourth points. TB calculates the point-to-point distance between the template and the segmentation result not only in the vertical direction like ED does, but also in the horizontal or temporal direction. TB is stricter and finds less number of patterns than ED and DTW. As shown in Figure 12(c), the amplitude and temporal distance between the segmentation result (blue) and the template are used to decide the classification result of TB. Therefore, the sequence given in Figure 12(a) is only found by ED as a "Cup with Handle" pattern while it was considered as a negative case by TB approach.

6.2.4. Patterns found by SVM

From the experiment results given in Table 2, six out of 22 "Cup with Handle" patterns found by SVM are not found by the other five approaches. Figure 13(a) and (b) show "Cup with Handle" patterns which are only found by SVM. Figure 13(c) shows that although the first point of segmentation result (blue) in Figure 13(a) is far lower than the template (red), SVM identifies this sequence as a "Cup with Handle" pattern.

6.2.5. Patterns found by HSMM

From the experiment results given in Table 2, all the patterns found by HSMM are also found by other pattern matching approaches.



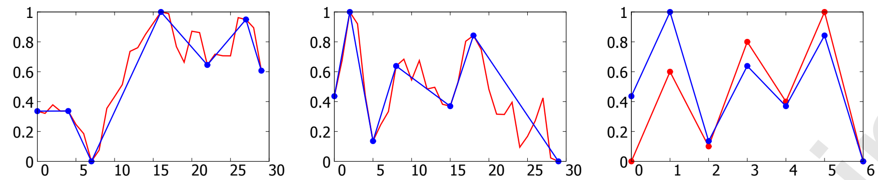
(a) (b) (c)

Figure 13: (a) The segmentation result (blue) on a normalized subsequence (red) from 9 March 2007 to 24 April 2007 in HSI found by SVM as a "Cup with Handle" pattern. (b) The segmentation result (blue) on a normalized subsequence (red) from 23 June 2010 to 3 August 2010 in HSI found by SVM as a "Cup with Handle" pattern. (c) The segmentation result in (a) (blue) and the template of the "Cup with Handle" pattern (red).

6.2.6. Patterns found by ANFIS

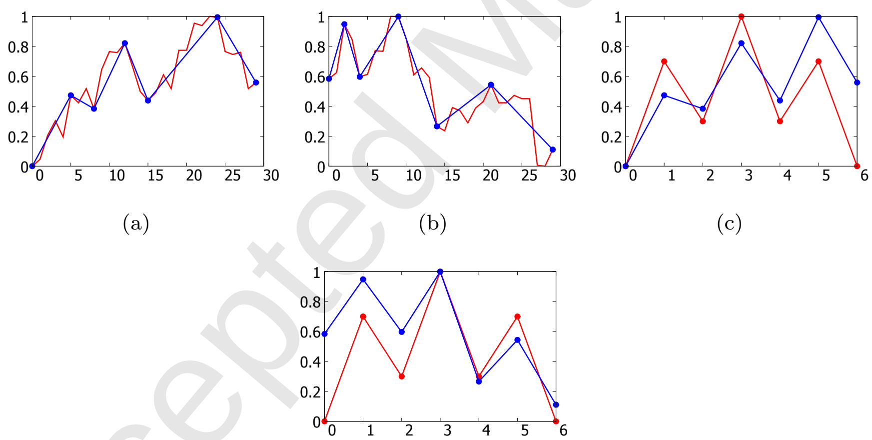
From the experiment results given in Table 2, 16 out of 28 "Wedge, Rising" patterns found by ANFIS are not found by any other five approaches. Four out of 23 "Head-and-Shoulders Tops" patterns found by ANFIS are not found by any other five approaches. Figure 14(a) and (b) show "Wedge, Rising" patterns which are only found by ANFIS. We can see in Figure 14(c) that the first peak of the segmentation result from Figure 14(b) is higher than the second peak which is dissimilar to the shape of the template of "Wedge, Rising" pattern (red). ANFIS recognizes it as a "Wedge, Rising" pattern although the first peak makes the shape of the segmentation result dissimilar to the template. Likewise, Figure 15(a) and (b) show two "Head-and-Shoulders Tops" patterns only found by ANFIS. Although the shapes of the segmentation results have one peak higher than the center peak when compared with the template (as shown in Figure 15(c) and (d)), ANFIS recognizes them as "Head-and-Shoulders Tops" patterns.

The pattern with the output which is closer to the expected output is more likely to be a qualified pattern. Table 3 shows some patterns found by ANFIS and the corresponding outputs. The column "Pattern" shows the patterns found by ANFIS. The expected outputs (or the label) of the "Wedge, Rising" pattern, the "Head-and-Shoulders Tops" pattern and the "Cup with Handle" pattern are 1, 2 and 3 respectively. The column "Output" shows the outputs of the ANFIS modle for the patterns found. In this paper, if the rounded output for a sequence is equal to the expected output, the sequence is recognized as the certain pattern. Figure 16(a) and (c) show two "Wedge, Rising" patterns found by ANFIS. The shape of the sequence in Figure 16(c) is more resemble to a "Wedge, Rising" because of the three high peaks. While the center peak is higher than the third



(a) (b) (c)

Figure 14: (a) The segmentation result (blue) on a normalized subsequence (red) from 19 December 2005 to 3 February 2006 in HSI found by ANFIS as a "Wedge, Rising" pattern. (b) The segmentation result (blue) on a normalized subsequence (red) from 31 January 2008 to 14 March 2008 in HSI found by ANFIS as a "Wedge, Rising" pattern. (c) The segmentation result (blue) from (b) and the template of the "Wedge, Rising" pattern (red).



(a) (b) (c) (d)

Figure 15: (a) The segmentation result (blue) on a normalized subsequence (red) from 4 June 2012 to 13 July 2012 in HSI found by ANFIS as a "Head-and-Shoulders Tops" pattern. (b) The segmentation result (blue) on a normalized subsequence (red) from 26 February 2013 to 9 April 2013 in HSI found by ANFIS as a "Head-and-Shoulders Tops" pattern. (c) The segmentation result from (a) (blue) and the template of the "Head-and-Shoulders Tops" pattern (red). (d) The segmentation result from (b) (blue) and the template of the "Head-and-Shoulders Tops" pattern (red).

Table 3: Patterns found by ANFIS and the corresponding outputs and expected outputs.

Pattern	Output	Expected output
Figure 16(a)	0.848841	1
Figure 16(c)	1.000455	1
Figure 15(a)	1.548064	2
Figure 17(a)	2.000654	2
Figure 18(a)	2.99995	3
Figure 18(c)	3	3

peak in Figure 16(a), the output of Figure 16(c) is closer to 1 than the output of Figure 16(a). Figure 15(a) and Figure 17(a) show two "Head-and-Shoulders Tops" patterns found by ANFIS. The output of Figure 17(a) is closer to 2 than the output of Figure 15(a). In Figure 17(a), the center head is taller than the two shoulders whereas in Figure 15(a), the right shoulder is taller than the center head. Likewise, Figure 18(a) and (c) show two "Cup with Handle" patterns found by ANFIS. The output of Figure 18(c) is closer to 3 than the output of Figure 18(a). The first point is higher than the second point in Figure 18(c) while the first point is lower than the second point in Figure 18(a).

6.2.7. Overlapped patterns found

From the experiment results given in Table 2, four out of 28 "Wedge, Rising" patterns found are also recognized as "Head-and-Shoulders, Tops" patterns for ANFIS. One out of two "Wedge, Rising" patterns found are also recognized as "Head-and-Shoulders Tops" patterns for TB. Two out of seven "Wedge, Rising" patterns found are also recognized as "Head-and-Shoulders, Tops" patterns for SVM. Two out of three "Wedge, Rising" patterns found are also recognized as "Head-and-Shoulders, Tops" patterns for HSMM. Eight out of 11 "Wedge, Rising" patterns found are also recognized as "Head-and-Shoulders Tops" patterns for ED. Ten out of 16 "Wedge, Rising" patterns found are also recognized as "Head-and-Shoulders Tops" patterns for DTW. ED and DTW can not clearly distinguish the "Wedge, Rising" pattern and the "Head-and-Shoulders Tops" pattern. Although RB approach is rigid and found less number of patterns, the patterns found by RB are distinguishable with each other. Figure 19(a) shows a pattern recognized by ANFIS, SVM, HSMM, ED and DTW as both a "Wedge, Rising" and a "Head-and-Shoulders Tops" pattern. Figure 19(b)

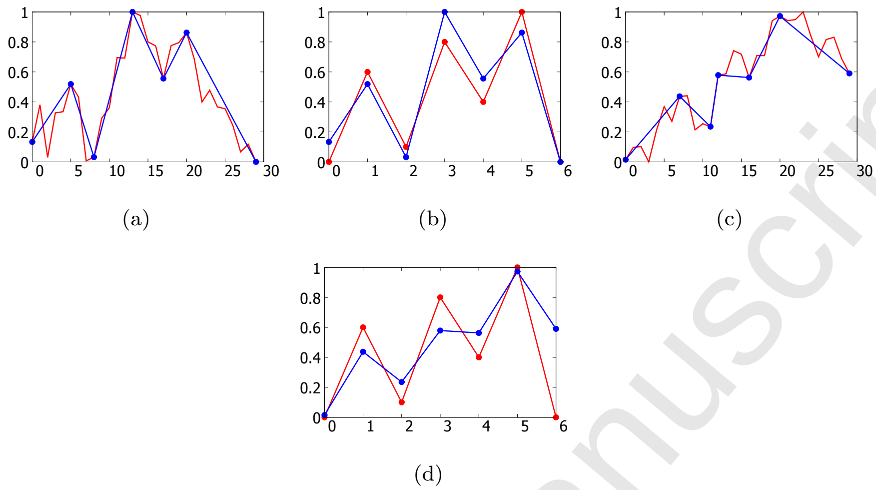


Figure 16: (a) The segmentation result (blue) on a normalized subsequence (red) from 4 July 2008 to 15 August 2008 in HSI found by ANFIS as a "Wedge, Rising" pattern. (b) The segmentation result from (a) (blue) and the template of the "Wedge, Rising" pattern (red). (c) The segmentation result (blue) on a normalized subsequence (red) from 17 March 2009 to 28 April 2009 in HSI found by ANFIS as a "Wedge, Rising" pattern. (d) The segmentation result from (c) (blue) and the template of the "Wedge, Rising" pattern (red).

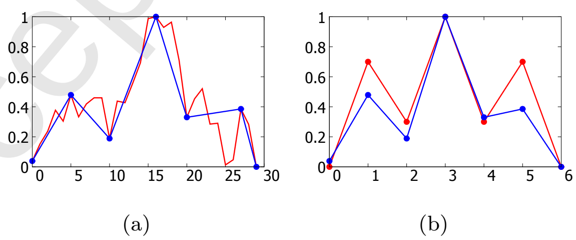
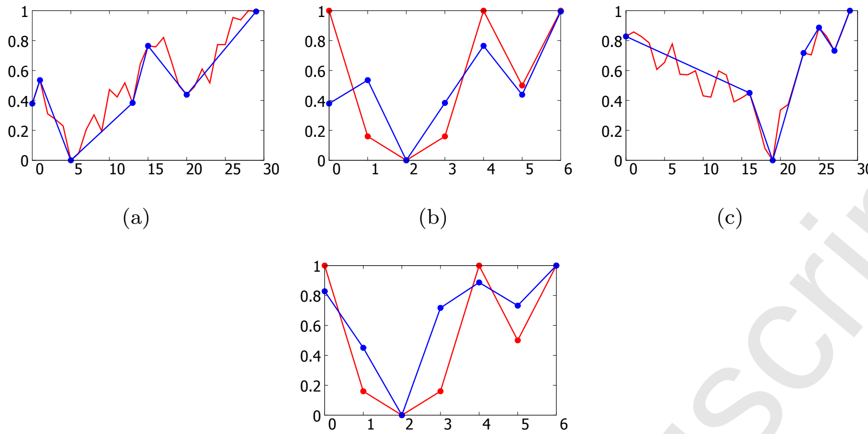


Figure 17: (a) The segmentation result (blue) on a normalized subsequence (red) from 8 October 2007 to 16 November 2007 in HSI found by ANFIS as a "Head-and-Shoulders Tops" pattern. (b) The segmentation result from (a) (blue) and the template of the "Head-and-Shoulders Tops" pattern (red).



(a) (b) (c)

(d)

Figure 18: (a) The segmentation result (blue) on a normalized subsequence (red) from 28 May 2012 to 6 July 2012 in HSI found by ANFIS as a "Cup with Handle" pattern. (b) The segmentation result from (a) (blue) and the template of the "Cup with Handle" pattern (red). (c) The segmentation result (blue) on a normalized subsequence (red) from 23 July 2007 to 31 August 2007 in HSI found by ANFIS as a "Cup with Handle" pattern. (d) The segmentation result from (c) (blue) and the template of the "Cup with Handle" pattern (red).

shows a pattern found by ANFIS, SVM, HSMM, TB, ED and DTW as both a "Wedge, Rising" pattern and a "Head-and-Shoulders Tops" pattern. The two confusing patterns in Figure 19(a) and (b) are similar to the "Wedge, Rising" and "Head-and-Shoulders Tops" templates from Figure 19(c), (d), Figure 19(e) and (f) to some extent.

7. Conclusion

In this paper, a novel application of ANFIS model for chart patterns matching in financial time series is presented. In addition, we propose a method to determine thresholds for classifying different chart patterns. ANFIS usex fuzzy if-then rules to describe different chart patterns. The non-linearity and structured knowledge representation are the primary advantages of ANFIS. In each rule, the distributions of the inputs are characterized by membership functions which are used to describe the deformations of the chart pattern templates. Therefore ANFIS can recognize deformations of a chart pattern. In addition, ANFIS can be easily applied in multiple classes classification problem by assigning different labels to different classes.

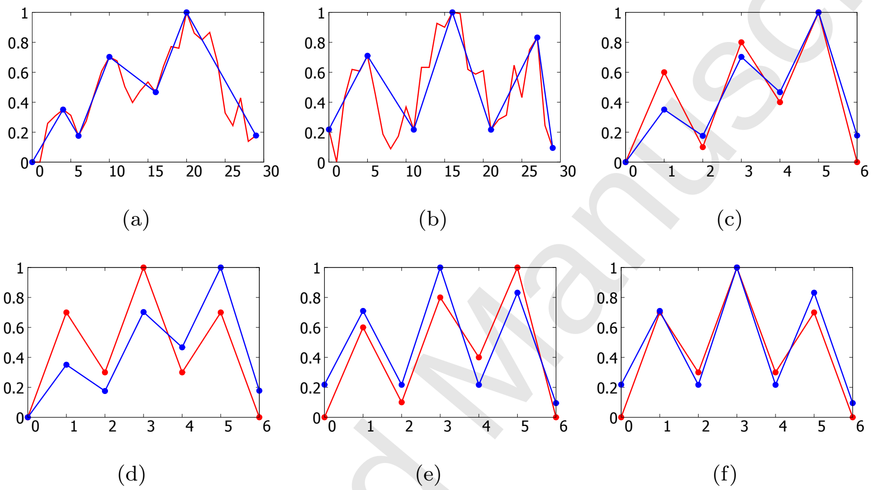


Figure 19: (a) The segmentation result (blue) on a normalized subsequence (red) from 4 April 2006 to 19 May 2006 in HSI found by ANFIS, SVM, HSMM, ED and DTW as a "Wedge, Rising" pattern and a "Head-and-Shoulders Tops" pattern. (b) The segmentation result (blue) on a normalized subsequence (red) from 13 June 2012 to 24 July 2014 in HSI found by found by ANFIS, SVM, HSMM, TB, ED and DTW as a "Wedge, Rising" pattern and a "Head-and-Shoulders Tops" pattern. (c) The segmentation result from (a) (blue) and the templates of the "Wedge, Rising" pattern (red). (d) The segmentation result from (a) (blue) and the template of the "Head-and-Shoulders Tops" pattern (red). (e) The segmentation result from (b) (blue) and the templates of the "Wedge, Rising" pattern (red). (f) The segmentation result from (b) (blue) and the template of the "Head-and-Shoulders Tops" pattern (red).

In the experiments, by using thresholds, ANFIS is able to recognize the negative cases. Therefore, ANFIS model only needs to be trained with the positive cases whereas SVM needs to be trained with both positive and negative cases. The experiment on synthetic data shows that ANFIS model takes more time to train and test than remaining five approaches, although their accuracies are similar (except the RB approach). ANFIS requires more time for training due to the application of gradient descent learning algorithm. RB is efficient since it checks a set of predefined rules which use the relative positions of the points from chart patterns. TB and ED are efficient since they perform simple distance calculations. Although the calculation of DTW is more complex than ED, it is efficient when the sequence is segmented beforehand.

In the experiment of real dataset, the shape of the patterns found by TB approach conforms to the template, for TB calculates the amplitude distance and temporal distance between the template and the sequence. RB approach classifies patterns according to the predefined rules and hence the patterns found are conformed with the descriptions of the rules. Since ANFIS and SVM are trained to recognize deformations of the chart patterns, more deformations of the patterns are found with these two approaches. ANFIS found more "Wedge, Rising" patterns and "Head-and-Shoulders Tops" patterns than SVM, while SVM found more "Cup with Handle" patterns than ANFIS. All the "Cup with Handle" patterns found by ANFIS are also found by SVM. DTW focuses on one specific kind of deformation of the chart patterns, that is, the deformations that stretch in time. Experimental results reveal that the patterns found by DTW can highly dissimilar to the template. If one wants to find some patterns that varies from the template slightly, TB and RB are good choices. If one can accept high deformation and would like to find as many patterns as possible, ANFIS is a good choice.

In 6.2.7, although RB found less number of patterns than other approaches, it can well distinguish different kinds of chart patterns by the predefined rules. For ED and DTW, most of the "Wedge, Rising" and "Head-and-Shoulders Tops" patterns found are resemble to each other. We find that ED and DTW are not able to clearly distinguish this two kinds of patterns. Compared with ED and DTW, ANFIS and SVM can distinguish most of the chart patterns found.

For ANFIS, one of the conditions to accept a sequence as a certain chart pattern is that the rounded output for the sequence is equal to the expected output of the certain chart pattern. The patterns found may have different outputs but the same rounded outputs. With ANFIS, we can quantify the difference of the same kind of patterns found through the outputs. The pattern with the output which is closer to the expected output is more likely to be a qualified pattern. As a future work, we are planning to tune this condition of accepting a sequence. For example, we can determine a threshold for the outputs as the condition to accept a sequence rather than use the rounded output to compare with the expected output.

Acknowledgement

This research was funded by the Research Committee of University of Macau, grant MYRG2015-00054-FST and MYRG2016-00148-FST.

Appendix A

Table 4 shows the classification of chart patterns. The patterns used in the experiments are highlighted with asterisks.

[1] R. D. Edwards, J. Magee, W. Bassetti, Technical analysis of stock trends, CRC Press, 2007.

[2] T. N. Bulkowski, Encyclopedia of chart patterns, 2nd Edition, John Wiley & Sons, 2011.

[3] T.-c. Fu, F.-l. Chung, R. Luk, C.-m. Ng, Stock time series pattern matching: Template-based vs. rule-based approaches, Engineering Applications of Artificial Intelligence 20 (3) (2007) 347–364.

[4] D. J. Berndt, J. Clifford, Using dynamic time warping to find patterns in time series., in: KDD workshop, Vol. 10, Seattle, WA, 1994, pp. 359–370.

[5] X. Ge, P. Smyth, Deformable markov model templates for time-series pattern matching, in: Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2000, pp. 81–90.

[6] S. Kim, P. Smyth, Segmental hidden markov models with random effects for waveform modeling, The Journal of Machine Learning Research 7 (2006) 945–969.

[7] A. P. Dempster, N. M. Laird, D. B. Rubin, Maximum likelihood from incomplete data via the em algorithm, Journal of the royal statistical society. Series B (methodological) 39 (1) (1977) 1–38.

Table 4: Classification of chart patterns.

C1: Patterns with fluctuations (variable-fluctuations)	C3: Patterns with curves
Broadening Bottoms	Bump-and-Run Reversal Bottoms
Broadening Formations, Right-Angled and Ascending	Bump-and-Run Reversal Tops
Broadening Formations, Right-Angled and Descending	Cup with Handle*
Broadening Tops	Cup with Handle, Inverted
Broadening Wedges, Ascending	Double Bottoms, Adam and Eve
Broadening Wedges, Descending	Double Bottoms, Eve and Adam
Flags	Double Bottoms, Eve and Eve
Flags, High and Tight	Double Tops, Adam and Eve
Head-and-Shoulders Bottoms, Complex	Double Tops, Eve and Adam
Head-and-Shoulders Tops, Complex	Double Tops, Eve and Eve
Pennants	Rounding Bottoms
Rectangle Bottoms	Rounding Tops
Rectangle Tops	Scallops, Ascending
Triangles, Ascending	Scallops, Ascending and Inverted
Triangles, Descending	Scallops, Descending
Triangles, Symmetrical	Scallops, Descending and Inverted
Wedges, Falling	
Wedges, Rising*	
Diamond Bottoms	
Diamond Tops	
C2: Patterns with fluctuations (fixed-fluctuations)	C4: Patterns with spikes
Double Bottoms, Adam and Adam	Horn Bottoms
Double Tops, Adam and Adam	Horn Tops
Head-and-Shoulders Bottoms	Pipe Bottoms
Head-and-Shoulders Tops*	Pipe Tops
Measured Move Down	
Measured Move Up	C5: Patterns with gaps
Three Falling Peaks	Gaps
Three Rising Valleys	Island Reversals
Triple Bottoms	Islands, Long
Triple Tops	

[8] L. R. Rabiner, A tutorial on hidden markov models and selected applications in speech recognition, Proceedings of the IEEE 77 (2) (1989) 257–286.

[9] J.-S. R. Jang, Anfis: adaptive-network-based fuzzy inference system, Systems, Man and Cybernetics, IEEE Transactions on 23 (3) (1993) 665–685.

[10] J.-S. R. Jang, C.-T. Sun, Neuro-fuzzy modeling and control, Proceedings of the IEEE 83 (3) (1995) 378–406.

[11] S. L. Chiu, Fuzzy model identification based on cluster estimation, Journal of Intelligent & Fuzzy Systems 2 (3) (1994) 267–278.

[12] J. B. Mbede, P. Ele, C.-M. Mveh-Abia, Y. Toure, V. Graefe, S. Ma, Intelligent mobile manipulator navigation using adaptive neuro-fuzzy systems, Information sciences 171 (4) (2005) 447–474.

[13] P. Melin, O. Castillo, Intelligent control of a stepping motor drive using an adaptive neuro–fuzzy inference system, Information Sciences 170 (2) (2005) 133–151.

[14] S. Y. Belal, A. F. G. Taktak, A. J. Nevill, S. A. Spencer, D. Roden, S. Bevan, Automatic detection of distorted plethysmogram pulses in neonates and paediatric patients using an adaptive-network-based fuzzy inference system, Artificial Intelligence in Medicine 24 (2) (2002) 149–165.

[15] E. D. Übeylı, I. Güler, Automatic detection of erythemato-squamous diseases using adaptive neuro-fuzzy inference systems, Computers in Biology and Medicine 35 (5) (2005) 421–433.

[16] I. Güler, E. D. Übeyli, Application of adaptive neuro-fuzzy inference system for detection of electrocardiographic changes in patients with partial epilepsy using feature extraction, Expert Systems with Applications 27 (3) (2004) 323–330.

[17] I. Güler, E. D. Übeyli, Adaptive neuro-fuzzy inference system for classification of eeg signals using wavelet coefficients, Journal of neuroscience methods 148 (2) (2005) 113–121.

[18] F.-L. Chung, T.-C. Fu, R. Luk, V. Ng, Flexible time series pattern matching based on perceptually important points, in: International joint conference on artificial intelligence workshop on learning from temporal and spatial data, 2001, pp. 1–7.

[19] E. J. Keogh, M. J. Pazzani, A simple dimensionality reduction technique for fast similarity search in large time series databases, in: Knowledge Discovery and Data Mining. Current Issues and New Applications, Springer, 2000, pp. 122–133.

[20] E. Keogh, S. Chu, D. Hart, M. Pazzani, An online algorithm for segmenting time series, in: Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on, IEEE, 2001, pp. 289–296.

[21] Y.-W. Si, J. Yin, Obst-based segmentation approach to financial time series, Engineering Applications of Artificial Intelligence 26 (10) (2013) 2581–2596.

[22] Y. Wan, X. Gong, Y.-W. Si, Effect of segmentation on financial time series pattern matching, Applied Soft Computing 38 (2016) 346–359.

[23] Z. Xing, J. Pei, E. Keogh, A brief survey on sequence classification, ACM SIGKDD Explorations Newsletter 12 (1) (2010) 40–48.

[24] C.-C. Chang, C.-J. Lin, Libsvm: A library for support vector machines, ACM Transactions on Intelligent Systems and Technology (TIST) 2 (3) (2011) 27.

[25] S.-Z. Yu, Hidden semi-markov models, Artificial Intelligence 174 (2) (2010) 215–243.

[26] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436–444.

[27] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural computation 9 (8) (1997) 1735–1780.

[28] S. Liu, N. Yang, M. Li, M. Zhou, A recursive recurrent neural network for statistical machine translation.

[29] T. Mikolov, M. Karafiát, L. Burget, J. Cernockỳ, S. Khudanpur, Recurrent neural network based language model., in: Interspeech, Vol. 2, 2010, p. 3.

[30] K.-P. Chan, A. W.-C. Fu, Efficient time series matching by wavelets, in: Data Engineering, 1999. Proceedings., 15th International Conference on, IEEE, 1999, pp. 126–133.

[31] Z. R. Struzik, A. Siebes, The haar wavelet transform in the time series similarity paradigm, in: European Conference on Principles of Data Mining and Knowledge Discovery, Springer, 1999, pp. 12–22.

[32] J. N. Liu, R. W. Kwong, Automatic extraction and identification of chart patterns towards financial forecast, Applied Soft Computing 7 (4) (2007) 1197–1208.

[33] T. Takagi, M. Sugeno, Fuzzy identification of systems and its applications to modeling and control, Systems, Man and Cybernetics, IEEE Transactions on (1) (1985) 116–132.

[34] M. Buragohain, C. Mahanta, A novel approach for anfis modelling based on full factorial design, Applied Soft Computing 8 (1) (2008) 609–625.

[35] Z. Zhang, J. Jiang, X. Liu, R. Lau, H. Wang, R. Zhang, A real time hybrid pattern matching scheme for stock time series, in: Proceedings of the Twenty-First Australasian Conference on Database Technologies-Volume 104, Australian Computer Society, Inc., 2010, pp. 161–170.

[36] T. M. Inc, Matlab fuzzy logic toolbox, http://www.mathworks.com/products/fuzzy-logic/, [Accessed: 2-August-2016] (1994-2016).

[37] Yahoo, Yahoo finance, https://ca.finance.yahoo.com/q/hp?s=YHOO, [Accessed: 23-Mar-2016] (2009).

- A novel application of ANFIS for chart patterns matching is presented.
- We propose a method to determine thresholds in ANFIS for classification.
- Proposed approach is tested on diverse classes of known patterns in financial time series.
- Experiments show that ANFIS model is comparable with the SVM approach.

*Graphical abstract (for review)