Soft Comput (2016) 20:1991–2019
DOI 10.1007/s00500-015-1619-3

METHODOLOGIES AND APPLICATION

A novel multivariate fuzzy time series based forecasting algorithm incorporating the effect of clustering on prediction

Arunava Roy

Published online: 25 February 2015

Abstract Forecasting has often played predominant roles in daily life as necessary measures can be taken to bypass the undesired and detrimental future prompted by this fact, the issue of forecasting becomes one of the most important topics of research for the modern scientists and as a result several innovative forecasting techniques have been developed. Amongst various well-known forecasting techniques, fuzzy time series-based methods are successfully used, though they are suffering from some serious drawbacks, viz., fixed sized intervals, using some fixed membership values (0, 0.5, and 1) and moreover, the defuzzification process only deals with the factor that is to be predicted. Additionally, most of the existing and widely used fuzzy time series-based forecasting algorithms employ their own clustering techniques that may be data-dependent and in turn the predictive accuracy decrease. Prompted by the fact, the present author developed a novel multivariate fuzzy forecasting algorithm that is able to remove all the drawbacks as also can predict the future occurrences with better predictive accuracy. Moreover, the comparisons with the thirteen other existing frequently used forecasting algorithms (viz., conventional, fuzzy time series-based algorithms and ANN) were performed to demonstrate its better efficiency and predictive accuracy. Towards the end, the applicability and predictive accuracy of the developed algorithm has been demonstrated using three different data sets collected from three different domains, such as: oil agglomeration process (coal washing technique), frequently occurred web error prediction and the financial forecasting. The real dataset related to oil agglomeration was collected from CIMFER, Dhanbad, India, that regarding the frequently occurred web error codes of www.ismdhanbad.ac.in, the official website of ISM Dhanbad, was collected from the Indian School of Mines (ISM) Dhanbad, India server and the finance data set was collected from the Ministry of Statistical and Program Implementation (Govt. of India).

Keywords Fuzzy time series · Forecasting · Clustering · Fuzzy forecasting · Local influence · Global influence

Communicated by V. Loia.

A. Roy (✉)
Intelligent Security Systems Research Lab, Department of Computer Science, University of Memphis, Memphis, TN 38111, USA
e-mail: royism.arunava@gmail.com

List of symbols

Z^+	The set of positive integers
$cl(r)(r \in Z^+)$	Cluster r generated by the chosen clustering algorithm
$a_i(i \in Z^+)$	The ith ($i \in Z^+$) cluster or interval related to the main factor
$A_i(i \in Z^+)$	The linguistic variables corresponding to a_i
$b_{j,i}(i, j \in Z^+)$	The ith cluster or interval of the jth secondary factor
$B_{j,i}(i, j \in Z^+)$	The linguistic variables corresponding to $b_{j,i}$
$M(i, j)$	The jth($i \in Z^+$) element of the ith($i \in Z^+$) cluster of the main factor
$S(i)_{p,q}(i, p, q \in Z^+)$	The qth element of the pth cluster of the ith secondary factor
Predicted (i)	The predicted value of i

1 Introduction

In quite a many real-world applications related to science, technology, stock price forecasting, university enrollments,

weather forecasting, etc., prediction may often play a preeminent part as it can save peoples' precious time and imperative measures can well be taken prior getting some cynical results. Many widely used crisp and fuzzy set theory-based methods for predictions are available in literature (Aladag et al. 2008; Aliev et al. 2008; Bulut 2014; Bulut et al. 2012; Chen et al. 2013; Chen and Tanuwijaya 2011; Chen 1996; Chatterjee and Roy 2014a, b; Duru 2010, 2012; Duru and Bulut 2014; Dunn 1973; Huarng 2001a, b; Huarng and Yu 2005, 2006; Mamdani 1977; Ross 2010; Song and Chissom 1993a, b, 1994; Tanaka 1996; Tseng et al. 2001; Tsaur 2008; Khashei 2012; Zadeh 1975); of them the latter one has the capability of handling the uncertainty efficiently. Among several fuzzy set theory-based prediction techniques, the models exploiting the potentiality of fuzzy time series are the main subject of interest of the present paper as it has huge application in different aforementioned areas.

Fuzzy time series, based on the concept of fuzzy reasoning (Rabiei et al. 2014) proposed by Mamdani (1977), was first introduced by Song and Chissom (1993a, b, 1994), following which many variations and versions of fuzzy time series and their rigorous applications were discussed and published in many papers by several researchers (Aladag et al. 2008; Chen et al. 2013; Chen and Tanuwijaya 2011; Chatterjee and Roy 2014a, b; Chen 1996; Duru and Bulut 2014; Dunn 1973; Huarng 2001a, b; Huarng and Yu 2005, 2006; Mamdani 1977; Ross 2010; Song and Chissom 1993a, b, 1994; Tanaka 1996; Tseng et al. 2001). In their extensive study Song and Chissom (1993a, b, 1994) have mainly developed both the time invariant and time variant time series models and explained them with the help of some real-life examples, which are, however, improved by Chen (1996) and Chen and Tanuwijaya (2011) for the development of certain prediction algorithm. Another significant improvisation was made by Aladag et al. (2008) by using the feed forward neural network to define fuzzy relations in higher order fuzzy time series. Apart from this, several modifications of their work are found in literature. In a recent study, Chen and Tanuwijaya (2011) have adopted some relevant steps to make the interval size variable whereas, no improvisation was made to generate different membership values of the elements apart from 0, 0.5, and 1 as also the influences of the other factors are still not considered in the defuzzification process.

However, most of the existing extensively utilized fuzzy forecasting methods based on fuzzy time series used the static length of intervals, i.e., the same length of intervals, however, Huarng (2001a, b) pointed out that the lengths of intervals will greatly affect forecasting results, where the drawback of the static length of intervals being that the historical data are roughly put into the intervals, even if the variance of the historical data is not quite high. And the most important yet, the predictive accuracy of the existing fuzzy time series-based forecasting techniques is usually not satisfactory. Additionally, defuzzification process of the main factor does not consider the effects of the remaining secondary factors, which is, however, a major drawback of the existing widely used a forestated approaches. Furthermore, the membership distribution techniques employed by the aforementioned approaches suffer from some unrealistic assumptions and limitations in that the membership values can only be 0, 0.5, and 1. It is said to be an unrealistic assumption and limitation as, in many real-life applications, it is generally found that the membership of an element within a cluster (or interval) lies either in (0, 0.5) or (0.5, 1). In these cases, the predictive accuracy may be hampered if any of the membership values among 0, 0.5, and 1 is applied. Moreover, all the existing aforementioned fuzzy forecasting algorithms mainly utilize their own clustering techniques that may not be able to partition all the data sets correctly as the nature of the data sets change with the ever changing circumstances, due to which, frequently, it can be found that the data sets may contain categorical entries (true–false type), however, in the remaining cases it may contain numeric digits (real, integers) or mixture of numeric as well as categorical types (Bezdek et al. 1984; Dunn 1973; Hartingan and Wong 1979). Moreover, due to the changes in some statistical properties, viz., coefficient of variation, correlation, etc., the nature of the data sets may vary (Bezdek et al. 1984; Dunn 1973; Hartingan and Wong 1979). As a result, it is not possible for a single clustering algorithm to correctly partition (i.e., the partitions containing almost similar types of data) all the data sets. Consequently, the accuracy of the corresponding forecasting algorithm decreases significantly and it becomes data-dependent (Bezdek et al. 1984; Dunn 1973; Hartingan and Wong 1979), as the clustering algorithm used by them may be suitable for certain types of data sets. One possible solution to this problem is to search the particular clustering algorithms which are most suitable for the data sets under consideration. Apart from this, all the existing and frequently used fuzzy time series-based forecasting algorithms cannot check whether the data set is stationary or not, which can be a possible reason behind their unsatisfactory predictive accuracy. Hence, in the present paper, initially, the author checks whether the data set is stationary or not. If it is found to be stationary, continue with the proposed forecasting algorithm. Otherwise, the non-stationarity, trend components, etc. of the data set have been removed and next, the author selects the suitable clustering algorithm by checking the DVI (Bezdek et al. 1984; Dunn 1973; Hartingan and Wong 1979) index of the generated clusters which are frequently used to evaluate their quality.

From the foregoing survey of literature, it evinces itself clearly that the existing fuzzy time series-based forecasting techniques have the aforementioned limitations, which require certain improvements in the modeling technique for better predictive accuracy. Motivated by the aforestated com-

parative study, the present author improvised the multivariate fuzzy '*if-then*' rule-based model, incorporating both clustering (overlapping and non-overlapping) and prediction, by making the interval size variable (due to the selected suitable clustering algorithm) (described in Sect. 3), varying the membership value from 0 to 1 in contrast to the other researchers who have used only three values, viz., 0, 0.5, and 1, employing the influences of different factors at the time of defuzzification and prediction, and better accuracy for predicting different instances of the data set. The membership distribution technique adopted by the proposed fuzzy prediction algorithm is mainly based on some well-defined functions that can generate real numbers lying between 0 and 1, making it more realistic than others. Moreover, the defuzzification process acquired by the proposed algorithm can exploit the influences of different factors on the very factor that is to be predicted. Again, the selected suitable clustering algorithm (that depends on the nature of the data set) perfectly partition the data set and as a result, the forecasting accuracy increases (Huarng 2001a). Consequently, this improvised model as such is capable to overcome the aforesaid drawbacks.

Finally, three different examples related to three different areas of science and technology were cited, that demonstrate the potentiality and the applicability of the proposed algorithm over a vast domain. For the first example, the failure data, collected from the logs (access and error logs) of www.ismdhanbad.ac.in, the official website of ISM Dhanbad, India, was used and the above findings were satisfactorily validated. Quite on a contrary, the next example is related to the very popular oil agglomeration process for the beneficiation of coal fines (coal washing technique), a heavily used technique in the coal industries where the proposed algorithm proves its better predictive accuracy. The corresponding data were collected from the CIMFER (a CSIR Lab, Govt. of India), Dhanbad, India. Consequently, it can be concluded that the proposed algorithm has a vast area of implementation. The remaining example is related to a financial data collected from the Ministry of Statistics and Program Implementation, Govt. of India and the above findings are satisfactorily validated again. Next, the outcomes of the proposed algorithm were compared with various fuzzy and statistical techniques. Moreover, Chen and Tanuwijaya (2011) method was applied on the aforementioned examples by replacing its clustering technique with c-means (Bezdek et al. 1984) and k-means (Hartingan and Wong 1979) algorithms, respectively to demonstrate the influence of clustering on the predictive accuracy of the fuzzy time series-based forecasting models. Additionally, the proposed algorithm was validated by the 'Chi-square test of goodness of fit'.

Before proceeding to develop the fuzzy logic-based clustering and prediction algorithm, it would be apt for clarity to describe the organizational structure of the paper by discussing its important components in different sections and subsections. This includes review fuzzy time series in Sect. 2; development of the proposed algorithm in Sect. 3; discussion of the test results in Sect. 4. Towards the end, the important findings and conclusions of the present work are encapsulated in Sect. 5. Each of these sections is dealt with herein under in the paragraphs that follow:

2 Review of fuzzy time series

This subsection presents a review of the fuzzy time series.

Fuzzy time series-based on the concept of fuzzy reasoning proposed by Zadeh (1975), Mamdani (1977), was first introduced by Song and Chissom (1993a, b, 1994), following which many variations and versions of fuzzy time series and their rigorous applications were discussed and published in many papers by several researchers. Fuzzy time series is basically defined in the following way:

Definition 1 (*Fuzzy time series*) Assuming $Y(t), (t = 1, 2\ldots)$ is the subset of $\mathbb{R}^1$ (one-dimensional Euclidian space), which is the universe of discourse where fuzzy subsets $m_i(t), (i = 1, 2\ldots)$ are defined and let $F(t)$ be a collection of $m_i(t), (i = 1, 2\ldots)$, then, $F(t)$ is called a fuzzy time series defined on $Y(t)(t = 1, 2\ldots)$. Here, $F(t)$ is regarded as a linguistic variable and $m_i(t), (i = 1, 2\ldots)$ can be viewed as possible linguistic values of $F(t)$, where $m_i(t), (i = 1, 2\ldots)$ are represented by fuzzy sets.

From this, it can be observed that $F(t)$ is a function of time t, i.e., the value of $F(t)$ being different at different times. According to Mamdani (1977) and Chen and Tanuwijaya (2011), Chen et al. (2013), if there exists a fuzzy relationship $R(t, t-1)$, such that $F(t) = F(t-1) \circ R(t, t-1)$, where '$\circ$' is the fuzzy Max–Min composition operator, then $F(t)$ is caused by $F(t-1)$. The relationship between $F(t)$ and $F(t-1)$ is denoted by: $F(t) \rightarrow F(t-1)$. For example, for $t = 2013$, the fuzzy relationship between $F(t-1)$ and $F(t)$ is given by $F(2012) \rightarrow F(2013)$. It is to be noted that the right-hand side of the fuzzy relation represents the future fuzzy set (forecast), its crisp counterpart being denoted as $Y(t)$.

It is very much significant to note that the main difference between fuzzy and conventional time series lies in the fact that the values of the former are fuzzy sets, while the values of the latter are the real numbers. As a corollary, it can be roughly assumed that a fuzzy set is a class with fuzzy boundaries.

Definition 2 (*n order fuzzy relations*) If $F(t)$ be a fuzzy time series and if $F(t)$ is caused by $F(t-1), F(t-2)\ldots F(t-n)$, i.e., the next state is caused by the current and its n previous states, then this fuzzy logical relationship (FLR) would be represented by:

$$F(t-n), \ldots F(t-2), F(t-1) \rightarrow F(t)$$

and is called the n order fuzzy time series. n order-based fuzzy time series models are referred to as the higher order models.

If for any time t, $F(t) = F(t-1)$ and $F(t)$ has only finite elements, then $F(t)$ is called a time invariant fuzzy time series, otherwise, it is called a time variant fuzzy time series.

Different relevant examples of fuzzy time series were cited by Song and Chissom (1993a, b, 1994) as also by Chen and Tanuwijaya (2011), Chen et al. (2013).

3 The proposed algorithm

In this section, the proposed multivariate fuzzy forecasting algorithm has been developed. But before attempting to develop the algorithm, it would be appropriate to briefly touch upon the existing fuzzy forecasting techniques, after which the developed predictive algorithm will be compared to the latter's for accuracy and predictability.

Most of the existing fuzzy forecasting techniques, in general, employ the following four steps (Aladag et al. 2008; Bulut 2014; Bulut et al. 2012; Chatterjee and Roy 2014a, b; Chen et al. 2013; Chen and Tanuwijaya 2011; Chen 1996; Duru 2012, 2010; Dunn 1973; Huarng 2001a, b; Huarng and Yu 2005, 2006; Mamdani 1977; Ross 2010; Song and Chissom 1993a, b, 1994; Tanaka 1996; Tseng et al. 2001; Zadeh 1975):

- **Step 1**: Partitioning the universe of discourse into intervals,
- **Step 2**: Fuzzifying the historical data,
- **Step 3**: Building fuzzy logical relationship and obtaining fuzzy logical relationship groups, and
- **Step 4**: Calculating the forecast output.

However, all the aforementioned fuzzy time series-based forecasting algorithms are not checking the stationarity of the data set and, as a consequence, the predictive accuracy of these algorithms reduces. With this in mind, in the present paper, the author has introduced a step called stationarity checking (Step 1) and using the above four steps as the basis, a novel and innovative fuzzy clustering and prediction algorithm is being attempted to be developed to enable one to forecast different instances of the data set. For reference, the flow chart of the proposed algorithm may be seen in the already depicted Fig. 1. This new algorithm has five steps, e.g., Step 1. Stationarity Checking; Step 2. Clustering; Step 3. Computation of different parameters of the proposed algorithm; Step 4. Distribution of membership; Step 5. Multivariate fuzzy forecasting algorithm. Now, the development of the proposed algorithm is being done as follows:

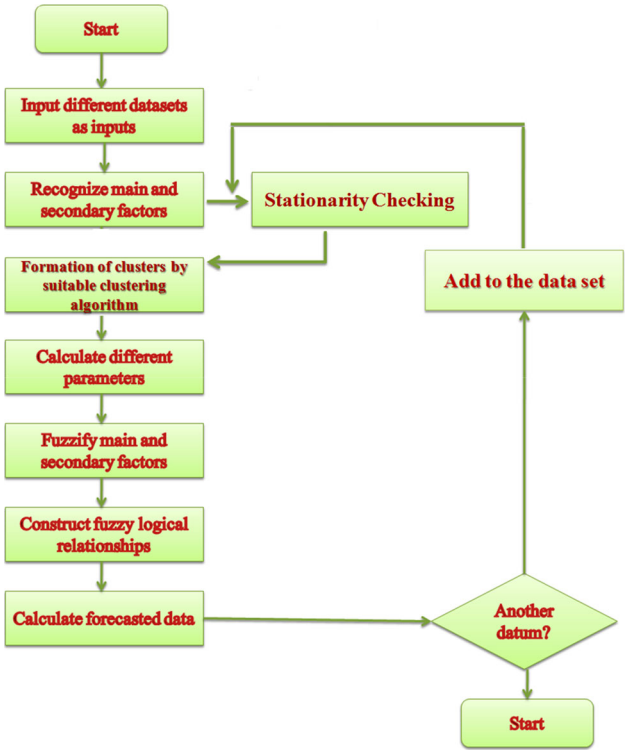

Fig. 1 Flow chart of the proposed multivariate fuzzy clustering algorithm

Step 1: stationarity checking

Data points are often non-stationary or have means, variances and covariances that change over time (Lutkepohl 2005). Non-stationary behaviors can be trends, cycles, random walks or combinations of the three. Non-stationary data, as a rule, are unpredictable and cannot be modeled or forecasted (Lutkepohl 2005). The results obtained using non-stationary time series may be spurious in that they may indicate a relationship between two variables where one does not exist. To receive consistent, reliable results, the non-stationary data needs to be transformed into stationary data (Lutkepohl 2005). In contrast to the non-stationary process that has a variable variance and a mean that does not remain near, or returns to a long-run mean over time, the stationary process reverts around a constant long-term mean and has a constant variance independent of time.

However, the sad fact is that a lot of important real-time series are not even approximately stationary. For example, most the share market data fall in this category. For example, most the share market data falls in this category. Hence, to check the stationarity of the input series, initially, the Dickey–Fuller test (Lutkepohl 2005) has been conducted. If the series is stationary, then, simply the suitable clustering algorithm has been selected for partitioning the data set. Otherwise,

the data set has been treated as the non-stationary time series (Lutkepohl 2005). The conventional approach is to try to separate time series like this into a persistent trend, and stationary fluctuations (or deviations) around the trend (Lutkepohl 2005),

$Y_t = X_t + Z_t$, i.e., series = fluctuations + trend.

Since a constant can be added or subtracted to each X_t without changing whether theyare stationary, then it can be stipulated $E(X_t) = 0$, i.e., $E(Y_t) = E(Z_t)$. In other situations, the decomposition might be multiplicative instead of additive, etc. (Lutkepohl 2005). Again, in case of multiple independent realizations $Y_{i,t}$ of the same process, say m of them having same trend Z_t, then the common trend can be found by averaging the time series:

$$Z_t = E(Y_{i,t}) \approx \sum_{i=1}^{m} Y_{i,t}$$

Multiple time series with the same trend do exist, especially in the experimental sciences (Lutkepohl 2005).

Once we have the fluctuations, and are reasonably satisfied that they're stationary, we can model them like any other stationary time series. Of course, to actually make predictions, the trend needs to be extrapolated, which is a harder business described as follows:

3.1 Trend components

The problem with making predictions when there is a substantial trend is that it is usually hard to know how to continue or extrapolate the trend beyond the last datapoint. If we are in the situation where we have multiple runs of the same process, we can at least extrapolate up to the limits of the different runs. If we have an actual model which tells us that the trend should follow a certain functional form, and we have estimated that model, we can use it to extrapolate (Lutkepohl 2005).

3.2 Pure random walk ($Y_t = Y_{t-1} + \varepsilon_t$)

Random walk predicts that the value at time "t" will be equal to the last period value plus a stochastic (non-systematic) component that is a white noise, which means ε_t is independent and identically distributed with mean "0" and variance "σ^2" (Lutkepohl 2005). Random walk can also be considered as a process integrated of some order, a process with a unit root or a process with a stochastic trend. It is a non-mean reverting process that can move away from the mean either in a positive or negative direction. Another characteristic of a random walk is that the variance evolves over time and goes to infinity as time goes to infinity and hence, a random walk cannot be predicted (Lutkepohl 2005).

3.3 Random walk with drift ($Y_t = \alpha + Y_{t-1} + \varepsilon_t$)

If the random walk model predicts that the value at time "t" will equal the last period's value plus a constant, or drift (α), and a white noise term (ε_t), then the process is random walk with a drift. It also does not revert to a long-run mean and has variance dependent on time (Lutkepohl 2005).

3.4 Deterministic trend ($Y_t = \alpha + \boldsymbol{\beta t} + \varepsilon_t$)

Often a random walk with a drift is confused for a deterministic trend. Both include a drift and a white noise component, but the value at time "t" in the case of a random walk is regressed on the last period's value (Y_{t-1}), while in the case of a deterministic trend it is regressed on a time trend (βt). A non-stationary process with a deterministic trend has a mean that grows around a fixed trend, which is constant and independent of time (Lutkepohl 2005).

3.5 Random walk with drift and deterministic trend ($Y_t = \alpha + Y_{t-1} + \boldsymbol{\beta t} + \varepsilon_t$)

Another example is a non-stationary process that combines a random walk with a drift component (α) and a deterministic trend (βt). It specifies the value at time "t" by the last period's value, a drift, a trend and a stochastic component (Lutkepohl 2005).

3.6 Seasonal components

Sometimes, it can be found that time series contain components which repeat, pretty exactly, over regular periods. These are called the seasonal components, after the obvious example of trends which cycle each year with the season. But they could cycle over months, weeks, days, etc... (Lutkepohl 2005). The decomposition of the process is thus

$$Y_t = X_t + Z_t + S_t,$$

where X_t can be considered as the stationary fluctuations, Z_t is the long-term trend and S_t is the repeating seasonal component. If $Z_t = 0$ or equivalently if we have a good estimate of it and can subtract it out, S_t can be found by averaging over multiple cycles of the seasonal trend (Lutkepohl 2005). Assume that, the period of the cycle is T, then $m = \frac{n}{T}$ number of full cycles can be found and S_t can be calculated as follows:

$$S_t \approx \frac{1}{m} \sum_{j=0}^{m-1} Y_{t+jT}.$$

This is because of the fact that, $Z_t = 0$, $Y_t = X_t + S_t$ and S_t is periodic, $S_t = S_{t+T}$. Sometimes, it is necessary to know the overall trend present in the data. If there are seasonal

components, they have to be subtracted before trying to find Z_t. The detrending can be done as follows:

Let Y_t has the linear time trend as follows:

$Y_t = \beta_0 + \beta t + X_t$

with X_t stationary. Then, if the difference between successive values Y_t has been taken, the trend goes away:

$Y_t - Y_{t-1} = \beta + X_t - X_{t-1}$.

Since, X_t is stationary, $(\beta + X_t - X_{t-1})$ is also stationary. However, if the first difference does not look stationary, then the other differences can be taken until the input series becomes stationary. In this way, the trend components can be removed from the data set. Similarly, applying the above procedure the random walk with or without a drift can be transformed to a stationary process (Lutkepohl 2005). Moreover, once $(Y_{t+1} - Y_t)$ has been predicted, Y_t can be added to get Y_{t+1}.

Step 2: clustering

(i) Initially, the suitable clustering algorithms [by checking the DVI (Bezdek et al. 1984; Dunn 1973; Hartingan and Wong 1979) indices of the generated clusters, given in Sect. 4] from the aforementioned classes apply to the data sets corresponding to the main (dependent variable) and the secondary factors (independent variables) to generate the variable-sized overlapping or non-overlapping clusters. In many cases, it can be found that different clustering algorithms are suitable for the data sets corresponding to the main and the secondary factors. Moreover, the number of clusters generated from the data sets of the main as well as the secondary factors may differ as it depends on the nature of the data set.

(ii) Next, the resulting clusters of the main and the secondary factors are related to that of the main factors for establishing the fuzzy logical relationships which has been discussed latter.

Step 3: computation of different parameters of the proposed algorithm

Some parameters of the proposed algorithm are now defined as below:

$\max[i]$, $(i \in Z^+)$: Maximum element of the ith cluster.

$\min[i]$, $(i \in Z^+)$: Minimum element of the ith cluster.

$$\text{mid}\,[i],\ (i \in Z^+) = 0.5 * (\max[i] + \min[i])$$

$$\text{mean} = \frac{1}{\text{no. of clusters of the main factor}} \sum_{i \in Z^+} \text{mid}\,[i] \quad (1)$$

$$\text{sum\_deviation} = \frac{1}{\text{no. of clusters of the main factor}} * \sqrt{\sum_{i \in Z^+} (\text{mean} - \text{mid}\,[i])^2} \quad (2)$$

$$\text{mean\_sec}\,[j] = \sum_{i \in Z^+} \left(\frac{\text{mid\_sec}[j]}{\text{no. of clusters of the } j\text{th secondary factor}} \right)$$

$$\text{sum\_deviation\_sec}[j] = \frac{1}{\text{no. of clusters of the } j\text{th secondary factor}} * \sqrt{\sum_{i \in Z^+} (\text{mean\_sec}\,[j] - \text{mid\_sec}\,[j]\,[i])^2}, \quad (3)$$

where $\text{mean\_sec}[j]$; $(j \in Z^+)$ is the mean_clust of the jth secondary factor and, as a consequence, k number of mean_sec can be found.

$$\text{global\_deviation} = \frac{1}{\prod_{k \in Z^+} \prod_{p \in Z^+} n_{k,p}} \sqrt{\sum_{i \in Z^+} (\text{mean} - \text{mid}\,[i])^2 + \sum_{j \in Z^+} \sum_{i \in Z^+} (\text{mean\_sec}\,[j] - mid\_sec\,[j]\,[i])^2}, \quad (4)$$

where $\text{mid\_sec}[j]\,[i]$; $(i, j \in Z^+)$ is the *mid* of the ith $(i \in Z^+)$ cluster of the jth$(j \in Z^+)$ secondary factor.

Here, $n_{k,p}$ is the total number of elements of the pth cluster of the kth main factor.

These parameters will be eventually used in the development of the algorithm.

Step 4: distribution of membership

In this step, it is to be necessarily checked up as to whether the distances between an element and the mid$[i]$, $(i \in Z^+)$ were less than *sum_deviation* or not. However, if it is less than *sum_deviation*, then the algorithm would itself generate a membership of the element in that cluster.

On the other hand, to check the influence of different secondary factors on the main factor, it is to be necessarily checked up as to whether the distances between the elements of the main factor and the $\text{mid\_sec}[j]\,[i]$; $(i, j \in Z^+)$ were less than *global_deviation* or not. If it is less, then the influence of the ith$(i \in Z^+)$ cluster of the jth$(j \in Z^+)$ secondary factor on the element of the main factor must be counted. This part enables the present forecasting algorithm to consider the influences of different factors on a particular factor.

Step 5: multivariate fuzzy forecasting algorithm

In this step, the multivariate fuzzy forecasting algorithm, based on the k-means clustering and fuzzy time series technique is developed. The novel feature of this algorithm is

that it takes care of overlapping as well as non-overlapping clusters.

(a) *Clustering* Let the universe of discourse of the main factor is divided into a number disjoint intervals or clusters (by the chosen clustering algorithm) denoted by a_i, $(i \in Z^+)$. The corresponding linguistic variables are denoted by A_i, $(i \in Z^+)$. Similarly, $b_{j,p}$, $(j, p \in Z^+)$ is the pth cluster of the jth secondary factor and the corresponding linguistic variable is denoted by $B_{j,p}$, $(j, p \in Z^+)$. In this paper, the dependent variable present in the system is the main factor; however, the independent variables are the secondary factors present in the system.

(b) *Defining fuzzy sets* The memberships of A_p in a_p (where $p \in Z^+$), i.e., the local influences (f_L), are determined by the following symmetric triangular fuzzy membership function that can remove the drawback of fixed membership values, viz., 0, 0.5 and 1 by taking more real numbers lying between 0 and 1:

$$f_L(A_p) = \begin{cases} 1; & \text{membership of } A_p \text{ in } a_p \\ \left(1 - \frac{x_i}{n_i * n_p}\right); & \text{where } i \in Z^+; i \neq p \\ 0; & \text{in case of the empty clusters of the main as well as the secondary factors.} \end{cases} \tag{5}$$

Again, the memberships of A_p in different clusters of the secondary factors $b_{j,i}$, $(i, j \in Z^+)$, i.e., the global influences (f_G), are determined by the following triangular membership function which can take more real numbers lying between 0 and 1 apart from 0, 0.5 and 1:

$$f_G(A_p) = \left(1 - \frac{x_{j,i}}{n_{j,i} * n_p}\right);\ i, j \in Z^+;\ p = \text{fixed} \tag{6}$$

Here, the memberships of the linguistic variable corresponding to a cluster of a particular factor on different clusters of the same factor are called the local influence (f_L). On the other hand, the memberships of the linguistic variable corresponding to a cluster of main factor on different clusters of the other factors are called the global influence (f_G). In this case, the significance of the local and the global influences are to establish the influences of several other clusters belong to the same or different factors on a cluster of a particular factor.

Variables used in the above equations are explained as follows:

n_p, $(p \in Z^+)$: Total number of elements of a_p, $(p \in Z^+)$.

x_i, $(0 \leq x_i \leq n_i * n_p)$: Total number of distances of the elements of a_i, $(i \in Z^+)$ from a_p, $(p \in Z^+)$ is greater than *sum_deviation* of the main factor (*Cf.* Step 3 above).

$x_{j,i}$, $(i, j \in Z^+)$: Total number of distances of the elements of a_i, $(i \in Z^+)$ from $b_{j,i}$, $(i, j \in Z^+)$ is greater than *global_deviation* (*Cf.* Step 3 above).

When all the distances are less than *sum_deviation* of the main factor, i.e., $x_i = 0$, it could well be discerned that $a_i = a_p$ and $f_L(A_p) = 1$.

For the fuzzy set representation of the linguistic variables A_p of main factor, both the local (f_L) and the global (f_G) influences are considered as follows:

$$A_p = \sum_{i \in Z^+} \frac{f_L(A_p)}{a_i} + \sum_{j \in Z^+} \sum_{i \in Z^+} \frac{f_G(A_p)}{b_{j,i}}. \tag{7}$$

Previously, it was mentioned that the secondary factors are mainly the independent variables present in the system. Hence, no other factors have the influences on the secondary factors. Consequently, in case of the secondary factors, the global influences are not considered. Now, the memberships of $B_{p,q}$ on $b_{p,i}$ can be defined with the help of the following triangular fuzzy membership function that can take more real numbers lying between 0 and 1 apart from 0, 0.5 and 1:

$$f(B_{p,q}) = \begin{cases} 1; & \text{membership of } B_{p,q} \text{ in } b_{p,q} \\ \left(1 - \frac{x_{p,i}}{n_{p,i} * n_{p,q}}\right); & \text{for other over lapping clusters;} \\ & p, q = \text{fixed}; i \in Z^+ \\ 0; & \text{for the empty clusters} \end{cases} \tag{8}$$

The fuzzy set representation of the linguistic variable $B_{p,q}$ is given as follows:

$$B_{p,q} = \sum_{i \in Z^+} \frac{f(B_{p,q})}{b_{p,i}}. \tag{9}$$

Variables used in the above equation are defined as follows:

$n_{p,q}$, $(p, q \in Z^+)$: Total number of elements in $b_{p,q}$, $(p, q \in Z^+)$.

$x_{p,i}$, $(0 \leq x_{p,i} \leq n_{p,i} * n_{p,q})$: Total number of distances of the elements of $b_{p,i}$, $(i \in Z^+)$ from $b_{p,q}$ is greater than *sum_deviation* of the pth secondary factor.

(c) *Prediction*: *Rule 1* The elements within the data set can be predicted by this rule. The membership of $M(p, q)$ on $M(i, j)$ (i.e., local influence, i.e., g_L, the influence of a particular occurrence of the main factor on the its other occurrence) can be defined with the help of the following triangular fuzzy membership function that can take more real numbers lying between 0 and 1 apart from 0, 0.5 and 1:

$$g_L(M(p,q)\_M(i,j)) = \left(1 - \frac{|M(p,q) - M(i,j)|}{\text{sum\_deviation}}\right);$$
$$|M(p,q) - M(i,j)| \leq \text{sum\_deviation} \neq 0 \tag{10}$$

Here, $M(p, q)$ is the qth element of the pth cluster of the main factor. Again, the memberships of $M(p, q)$ in $S(j)_{i,l}$,

i.e., the global influences (g_G), are determined by the following fuzzy triangular membership function:

$$g_G(M(p,q)\_S(j)_{i,l}) = \left(1 - \frac{\left|M(p,q) - S(j)_{i,l}\right|}{\text{global\_deviation}}\right);$$
$$\left|M(p,q) - S(j)_{i,l}\right| \leq \text{global\_deviation} \neq 0 \quad (11)$$

Variables used in the above equations are, in turn, defined as follows:

$g_L(M(p,q)\_M(i,j))$: local membership of M(p, q) on $M(i, j)$.

$g_G(M(p,q)\_S(j)_{i,l})$: global membership of M(p, q) on $S(j)_{i,l}$.

Next, the fuzzy sets corresponding to $M(p,q)$, $(p,q \in \mathrm{Z}^+)$ are then defined in the following manner:

$$M(p,q) = \sum_{i\in\mathrm{Z}^+}\sum_{j\in\mathrm{Z}^+}\frac{g_L(M(p,q)\_M(i,j))}{a_i} + \sum_{j\in\mathrm{Z}^+}\sum_{i\in\mathrm{Z}^+}\sum_{l\in\mathrm{Z}^+}\frac{g_G(M(p,q)\_S(j)_{i,l})}{b_{j,i}}. \quad (12)$$

Now in the next step, the construction of fuzzy logical relationship on fuzzified main and secondary factors is made as follows:

$$M(i,j), S(1)_{a,b}, S(2)_{c,d}, \ldots S(k)_{l,p} \rightarrow M(m,n),$$

where $M(i,j), S(1)_{a,b}, S(2)_{c,d}, \ldots S(k)_{l,p}$ denotes fuzzified value of the main factor, the fuzzified value of the first secondary factor, the fuzzified value of the second secondary factor, ..., and the fuzzified value of the kth secondary factor at stage t, then at the stage $(t+1)$ the main factor will be nth element of the mth cluster of the main factor.

The defuzzified predicted occurrences of the main factor can be calculated in the following manner:

$$\text{predicted}(M(p,q)) = \frac{1 * \text{mid}(a_p) + \sum_{i\in\mathrm{Z}^+}\left\{\sum_{j\in\mathrm{Z}^+}\left(1 - \frac{|M(p,q)-M(i,j)|}{\text{sum\_deviation}}\right)\right\} * \text{mid}(a_i) + \sum_{j\in\mathrm{Z}^+}\left\{\sum_{i\in\mathrm{Z}^+}\sum_{l\in\mathrm{Z}^+}\left(1 - \frac{|M(p,q)-S(j)_{i,l}|}{\text{global\_deviation}}\right)\right\} * \text{mid}(b_{j,i})}{1 + \sum_{j\in\mathrm{Z}^+}\left(1 - \frac{|M(p,q)-M(i,j)|}{\text{sum\_deviation}}\right) + \sum_{i\in\mathrm{Z}^+}\sum_{l\in\mathrm{Z}^+}\left(1 - \frac{|M(p,q)-S(j)_{i,l}|}{\text{global\_deviation}}\right)} \quad (13)$$

If the fuzzified value of the main factor, the fuzzified value of the first secondary factor, the fuzzified value of the second secondary factor, ..., and the fuzzified value of the kth secondary factor at time $(t-1)$ are $M(i,j), S(1)_{a,b}, S(2)_{c,d}, \ldots S(k)_{l,p}$, respectively, and there is a fuzzy logical relationship in the fuzzy logical relationship group, shown as follows:

$$M(i,j), S(1)_{a,b}, S(2)_{c,d}, \ldots S(k)_{l,p} \rightarrow M(m_1,n_1), M(m_2,n_2), M(m_3,n_3) \ldots M(m_r,n_r)$$

In case of the secondary factors (mainly the independent variables), the memberships of $S(i)_{p,q}$ on $S(i)_{l,j}$ (local influence) are defined with the help of the following triangular fuzzy membership function that can take more real numbers lying between 0 and 1 apart from 0, 0.5 and 1:

$$g(S(i)_{p,q}\_S(i)_{l,j}) = (1 - \frac{\left|S(i)_{p,q} - S(i)_{i,j}\right|}{\text{sum\_deviation\_sec}\,[i]});$$
$$|S(i)_{p,q} - S(i)_{i,j}| \leq \text{sum\_deviation\_sec}\,[i] \neq 0, \quad (14)$$

where $g(S(i)_{p,q}\_S(i)_{i,j})$ = Local membership of $S(i)_{p,q}$ on $S(i)_{i,j}$.

Fuzzy set representations for the elements of the secondary factors are defined as follows:

$$S(l)_{p,q} = \sum_{l\in\mathrm{Z}^+}\sum_{j\in\mathrm{Z}^+}\frac{g(S(i)_{p,q}\_S(i)_{l,j})}{b_{i,l}} \quad (15)$$

The secondary factors are mainly the independent variables present in the system. Hence, in this case, to form the fuzzy logical relationship the concept of univariate time series is used as follows:

$$S(l)_{p,q} \rightarrow S(l)_{r,s},$$

where $S(l)_{p,q} \rightarrow S(l)_{r,s}$ denotes that 'if the fuzzified value of the lth secondary factor at stage t is the qth element of its pth cluster, then at the stage $(t+1)$ the aforementioned secondary factor will be its sth element of the rth cluster'. This is because of the fact that, in this case no other factor except itself has the influence on an independent variable.

The defuzzified predicted occurrences of the lth$(l \in \mathrm{Z}^+)$ secondary factor can be calculated using the following equation:

$$\text{predicted}(S(l)_{p,q}) = \frac{\sum_{i\in\mathrm{Z}^+}\left\{\sum_{j\in\mathrm{Z}^+}\left(1 - \frac{|S(l)_{p,q}-S(l)_{i,j}|}{\text{sum\_deviation\_sec}[l]}\right)\right\} * \text{mid}(b_{l,i})}{\sum_{j\in\mathrm{Z}^+}\left(1 - \frac{|S(l)_{p,q}-S(l)_{i,j}|}{\text{sum\_deviation\_sec}[l]}\right)} \quad (16)$$

Rule 2 An important feature of this rule is that the elements lying outside the data set can precisely be predicted. The corresponding fuzzy logical relationship can be constructed as follows:

$M(i, j), S(1)_{p,q}, S(2)_{r,s}, \ldots S(n)_{x,y} \rightarrow \#$, where '#' is the element lying outside the data set. Then the predicted value of # can be calculated as follows:

$$\text{predicted}(\#) = \frac{\sum_{i\in Z^+} \text{mid}\,[a_i] + \sum_{j\in Z^+}\sum_{i\in Z^+}\left(1-\frac{x_{j,i}}{n}\right) * \text{mid\_sec}\left[b_{j,i}\right]}{\text{Total number of clusters of the main factor} + \sum_{j\in Z^+}\sum_{i\in Z^+}(1-\frac{x_{j,i}}{n})}, \tag{17}$$

where $x_{j,i}$ is the number of mid values of the clusters of the main factor having distances greater than the *global_deviation* from $\text{mid}\left[b_{j,i}\right]$ and n is total number of mid-values of the main factor.

The secondary factors are mainly the independent variables present in the system. Hence, in this case, to form the fuzzy logical relationship the concept of uni-variate time series is used as follows:

$$S(i)_{p,q} \rightarrow \#\_S(i),$$

where $S(i)_{p,q} \rightarrow \#\_S(i)$ denotes that 'if the fuzzified value of the ith secondary factor at stage t is the qth element of its pth cluster, then at the stage $(t+1)$ the aforementioned secondary factor will be defuzzified to its unknown value $\#\_S(i)$'. The defuzzified predicted value of '$\#\_S(j)$'(unknown occurrences of the ith secondary factor) can then be calculated as follows:

$$\text{predicted}(\#\_S(j)) = \frac{\sum_{i\in Z^+} \text{mid}\left[b_{j,i}\right]}{\text{Total number of clusters of the } j\text{th secondary factor}} \tag{18}$$

After this, the defuzzified unknown occurrences of the main and the secondary factors are inserted into the dataset and repeat the step 1 to step 4 of the developed algorithm to predict the next occurrences of them.

The above-developed algorithm can thus effectively handle both the overlapping and non-overlapping clusters, besides making predictions and handling the uncertainty as well.

4 Test results of the developed algorithm

In this section, three different real-life examples related to three different areas, viz., web technology, coal industries and finance were cited, that demonstrate the potentiality and the applicability of the proposed algorithm over a vast domain. The first example deals with the prediction of some frequently occurred web errors during the execution of www.ismdhanbad.ac.in, the official website for Indian School of Mines Dhanbad, India. Quite on a contrary, the next one deals with the prediction of the yield % of the clean coal during the oil agglomeration process for the beneficiation of coal fines. However, the last one deals with the prediction of records (mainly financial data), provided by the ministry of statistical and program implementation, Govt. of India.

The proposed forecasting method was compared with thirteen different conventional (uni-variate and multivariate time series models), e.g., VAR, MA, Holt-Winter, Box-Jenkins (Lutkepohl 2005), and fuzzy time series-based forecasting algorithms, viz., Bulut et al. (2012), Bulut (2014), Duru (2010, 2012), Chatterjee and Roy (2014a), Chatterjee and Roy (2014b), Chen and Tanuwijaya (2011) (replacing its clustering algorithm with c-means and k-means techniques, respectively). Moreover, the accuracy of the proposed algorithm has also been compared with the ANN approach (Aladag et al. 2008). To check the forecasting accuracy of the proposed algorithm, in this paper root mean squared error (RMSE), root median squared errors (RMdSE) and median relative absolute error (MdRAE) have been used as the accuracy metrics. RMSE is a biased accuracy metric (Hyndman 2006). On the other hand, RMdSE is not scale-free (Hyndman 2006). The comparative study can be found in Table 1, 5, 9, 11, 12, 13 and 14.

However, in many cases, it can be found that different clustering techniques are suitable for the main and different secondary factors due to the change in the nature or properties (viz., statistical, etc.) of the corresponding data sets. Hence, for simplicity of calculation, in this paper the author mainly concentrates on three very well-known clustering techniques, viz., c-means, k-means and the automatic clustering algorithm, to discuss the experimental results obtained by implementing the proposed algorithm. But prior going to discuss the experimental results, it would be apt for clarity to explain briefly the concept of DVI (Bezdek et al. 1984; Dunn 1973; Hartingan and Wong 1979) as it is used to check the quality of the generated clusters.

A validity index is used to evaluate the quality of the clusters generated by the clustering algorithm (Bezdek et al. 1984; Dunn 1973; Hartingan and Wong 1979). For the performance measure of the proposed algorithm and the quality of the generated clusters, in this paper, the Dynamic Validity Index (DVI) is used and are defined as follows:

Let n be the number of data points, k be the pre-defined upper bound of the number of clusters, and z_i be the center of the cluster c_i. The dynamic validity index (DVI) is given as follows:

$$\text{DVI} = \min_p \left\{\text{IntraRatio(p)} + \gamma * \text{InterRatio(p)}\right\}, \tag{19}$$

Table 1 DVI index values of different clusters of the data sets of the main factors

DVI		
c-means clustering	*k*-means clustering	Automatic clustering
Web data set		
0.069505004	0.00705373	2.001338869
Oil agglomeration data		
0.15009829	0.1312234796	2
Finance data		
1.99458593542	1.9201220134	2.0458593542

where IntraRatio and InterRatio are defined as follows:

$$\text{IntraRatio(p)} = \frac{\text{Intra(p)}}{\text{MaxIntra}}, \text{InterRatio(p)} = \frac{\text{Inter(p)}}{\text{MaxInter}}$$

$$\text{Intra}(p) = \frac{1}{n}\sum_{i=1}^{p}\sum_{x\in \text{C}_i} \text{x} - z_i^2, \text{MaxIntra} = \max_i \{\text{Intra}(i)\}$$

$$\text{Inter}(p) = \frac{\max\limits_{i,\, j} (z_i - z_j^2)}{\min\limits_{i \neq j} (z_i - z_j^2)} \sum_{i=1}^{p} \frac{1}{\sum_{j=1}^{p} z_i - z_j^2},$$

$$\text{MaxInter} = \max_i \{\text{Inter}(i)\}$$

For simplicity of calculation, in the present study, *k*-means, *c*-means (with 3 clusters each) and the automatic clustering algorithms are applied to partition the data sets. The DVI of the clusters of all the data sets, used in this study, generated by the aforementioned clustering algorithms are shown in Table 1. The forecasted values may change with the clustering algorithm.

4.1 An example regarding web error prediction

In this subsection, the developed algorithm has been validated by predicting the frequently occurred web errors using the data collected from the HTTP log files (error and access logs) (Huynh and Miller 2009) of http://www.ismdhanbad.ac.in, i.e., the official website of Indian School of Mines Dhanbad, India, which is a non-commercial and dynamic website that utilizes the PHP (http://www.php.net) scripting language, MySql (http://www.mysql.com) for the back-end database and is hosted on an Apache HTTP Daemon. For scrutinizing the stability and reliability of the data, the log files (HTTP access and error logs) were chosen to cover 387 consecutive days, starting from 30th September 2010 to 22nd October 2011, during which, the website had received approximately, 6,367,893 hits, 188,369 unique visitors, 13,612 unique URLs, 25,433 unique user agents [viz. Mozilla/5.0+ (compatible; +Googlebot/2.1; ++ http://www.google.com/bot.html)], transferred a total amount of 87,964,646 KBytes data, and approximately, 35,137 numbers of sessions were created. The most frequently-occurred web failures corresponding to each day of http://www.ismdhanbad.ac.inare tabulated and shown in Table 2. From this table, it is observed that, the error code 404 numerically dominates the others, which is in tune with the findings of the survey results from 1994 to 1998 by the Graphics, Visualization, and Usability Center of Georgia Institute of Technology (http://www.gvu.gatech.edu/user_surveys/), which states that 404 errors are most commonly occurred errors that users encounter while browsing the web. The other

Table 2 The occurrences of different frequently occurred error codes from 30/9/2010 to 22/10/2011 along with their positions in the respective clusters generated by the *k*-means clustering algorithm. Dickey–Fuller test for checking stationarity of the series corresponding to the main, 1st and 2nd secondary factors

Main factor (404)	1st secondary factor (406)	2nd secondary factor (403)
$3890 = M(1,1)$	$0 = S(1)_{1,1}$	$6 = S(2)_{1,1}$
$3646 = M(3,1)$	$4 = S(1)_{3,1}$	$4 = S(2)_{1,1}$
$2387 = M(1,2)$	$346 = S(1)_{3,2}$	$3 = S(2)_{1,2}$
$3852 = M(1,3)$	$342 = S(1)_{1,2}$	$3 = S(2)_{3,3}$
$3283 = M(1,4)$	$12 = S(1)_{1,3}$	$6 = S(2)_{2,4}$
$3215 = M(1,5)$	$0 = S(1)_{1,4}$	$14 = S(2)_{1,5}$
$3097 = M(3,2)$	$0 = S(1)_{3,3}$	$2 = S(2)_{2,1}$
$4875 = M(2,10)$	$342 = S(1)_{1,5}$	$10 = S(2)_{1,2}$
$2687 = M(2,4)$	$0 = S(1)_{1,6}$	$2 = S(2)_{3,3}$
$2791 = M(2,5)$	$0 = S(1)_{1,7}$	$6 = S(2)_{3,4}$
$2503 = M(2,2)$	$0 = S(1)_{2,1}$	$6 = S(2)_{2,5}$
$2415 = M(3,3)$	$318 = S(1)_{1,8}$	$14 = S(2)_{2,2}$
$4372 = M(1,6)$	$0 = S(1)_{1,9}$	$10 = S(2)_{1,6}$
$3199 = M(2,11)$	$0 = S(1)_{1,10}$	$3 = S(2)_{1,6}$
$2884 = M(2,6)$	$0 = S(1)_{1,11}$	$4 = S(2)_{3,7}$
$2809 = M(2,7)$	$0 = S(1)_{1,12}$	$6 = S(2)_{1,8}$
$2797 = M(2,8)$	$0 = S(1)_{3,4}$	$1 = S(2)_{2,9}$
$2940 = M(2,9)$	$343 = S(1)_{1,13}$	$9 = S(2)_{1,10}$
$2856 = M(1,7)$	$0 = S(1)_{1,14}$	$3 = S(2)_{1,7}$
$3465 = M(1,8)$	$4 = S(1)_{1,15}$	$4 = S(2)_{1,8}$
$3485 = M(1,9)$	$1 = S(1)_{1,16}$	$4 = S(2)_{3,11}$
$2694 = M(2,3)$	$0 = S(1)_{1,17}$	$6 = S(2)_{2,12}$
$2403 = M(2,1)$	$0 = S(1)_{1,18}$	$10 = S(2)_{1,13}$
…	…	…

Dickey–Fuller test

Series	*p* values	Result
Main factor (404)	$0.01 < 0.05$	Stationary
1st secondary factor (406)	$0.01 < 0.05$	Stationary
2nd secondary factor (403)	$0.01 < 0.05$	Stationary

two most frequently occurred web error codes in case of http://www.ismdhanbad.ac.in are 406 (not acceptable) and 403 (forbidden) (Huynh and Miller 2009). Therefore, the error code 404 is now considered as the main factor and the error codes 406, 403 are now treated as the two secondary factors. Accordingly, the occurrences of 404 (main factor) are preferentially predicted on the basis of 406 and 403 (secondary factors). Now, for the modeling purpose 70 % of the data set, i.e., (387 × 0.7) ≈ 270 data have been used and remaining 117 (30 %) have been left for prediction purpose (post sample period).

Again, the stationarity of the series of occurrences corresponding to the error codes 404, 406, 403 has also been checked using the Dickey–Fuller test (Lutkepohl 2005)and found that all the series are stationary (Table 2).

Initially, for the simplicity of calculation, the main and the secondary factors, shown in Table 2, are partitioned into clusters with the help the k-means (with 3 clusters), c-means (with 3 clusters), and the automatic clustering algorithm (Chen and Tanuwijaya 2011) and the corresponding results are shown in Table 3. Next, different clustering indices, viz., DVI, are calculated to check the quality of the generated clusters, and it is found that the k-means (with 3 clusters) is most suitable for the data set shown in Table 2 due to lowest DVI among all the aforementioned clustering algorithms. Consequently, the proposed algorithm employs the k-means clustering technique to partition the main and the secondary factors in step 2 (*Cf.* Sect. 3). The resulting non-overlapping, non-empty clusters of the main and the secondary factors are denoted by a_i; $(i = 1, 2, 3)$ and $b_{i,j}$; $(i = 1, 2;\ j = 1, 2, 3)$, respectively. The linguistic variables corresponding to the clusters of the main and the secondary factors are denoted by A_i; $(i = 1, 2, 3)$ and $B_{i,j}$; $(i = 1, 2;\ j = 1, 2, 3)$, respectively.

Next, using step 3 of the proposed algorithm (*Cf.* Sect. 3), different parameters are calculated and are shown in Table 3. The fuzzy sets corresponding to A_i; $(i = 1, 2, 3)$ and $B_{i,j}$; $(i = 1, 2;\ j = 1, 2, 3)$ are defined using step 4(b)(*Cf.* Sect. 3) of the developed algorithm. For example, the fuzzy set A_1 (linguistic variable corresponding to a_1) can be defined as follows [using Eq. (7)]:

$$A_1 = \frac{1}{a_1} + \frac{\left(1 - \frac{8}{8*12}\right)}{a_2} + \frac{\left(1 - \frac{1}{8*3}\right)}{a_3} + \sum_{j=1}^{2}\sum_{i=1}^{3} \frac{\left(1 - \frac{x_{j,i}}{n_{j,i}*n_p}\right)}{b_{j,i}} = \frac{1}{a_1} + \frac{0.917}{a_2} + \frac{0.958}{a_3} + \sum_{j=1}^{2}\sum_{i=1}^{3} \frac{0}{b_{j,i}}$$

From the above fuzzy set representation, it is quite clear that the membership values can be real numbers lying between 0 and 1, apart from 0,0.5 and 1. Moreover, from the above equation, it is quite clear that there are influences of different secondary factors on the main factor, which is, however, a remarkable feature of the proposed algorithm. Similarly, the fuzzy sets corresponding to the remaining main, 1st and 2nd secondary factors can be calculated. From Table 2, it is found that 3870 = M(1, 1) ∈ a_1. Similarly, the positions of the other elements of the main and the secondary factors can be found.

Again, sum_deviation_sec[1] = 3.623 and sum_deviation_ sec[2] = 0.7577. The fuzzy set representation for $B_{1,3}$ (linguistic variable corresponding to $b_{1,3}$) is given as follows:

$$B_{1,3} = \sum_{i=1}^{3} \frac{f(B_{1,3})}{b_{1,i}},$$

$$\text{or, } B_{1,3} = \frac{1}{b_{1,3}} + \frac{\left(1 - \frac{4}{20}\right)}{b_{1,1}} + \frac{\left(1 - \frac{5}{10}\right)}{b_{1,2}} = \frac{1}{b_{1,3}} + \frac{0.8}{b_{1,1}} + \frac{0.5}{b_{1,2}}.$$

Next, different fuzzified occurrences of the main, 1st, and 2nd secondary factors of www.ismdhanbad.ac.in were tabulated and the data are shown in Table 2. Using Rule 1 of the developed algorithm, different fuzzy logical relationships were established. From Table 2, it is seen that 3,205 = $M(1, 5) \in a_1$. Accordingly, the corresponding fuzzy set can be defined, using Eq. (13), as given below:

$$M(\mathrm{p}, \mathrm{q}) = \sum_{i \in Z^+} \sum_{j \in Z^+} \frac{g_L(M(1,5)\_M(i,j))}{a_i} + \sum_{j \in Z^+} \sum_{i \in Z^+} \sum_{l \in Z^+} \frac{g_G(M(1,5)\_S(j)_{i,l})}{b_{j,i}}$$

In the same way, fuzzy sets corresponding to the other fuzzified occurrences of the main and secondary factors were defined. The fuzzy set representation for $4 = S(1)_{3,1}$ is given as follows:

$$S(1)_{3,1} = \sum_{i=1}^{3} \sum_{j \in Z^+} \frac{g_L(S(1)_{3,1}\_S(1)_{i,j})}{b_{1,i}}$$

Again, from first row of Table 2 using Rule 1 (*Cf.* Sect. 3), the following fuzzy logical relationship was formed.

'If the 1st element of the 1st cluster of the main factor (i.e., fuzzified value $M(3,1)$), 1st element of the 1st cluster of the 1st secondary factor (i.e., fuzzified value $S(1)_{1,1}$), and 1st element of the 1st cluster of the 2nd secondary factor (i.e., fuzzified value $S(2)_{1,1}$) are at stage 1, then at the stage 2 the main factor will be the 3rd element of the 1st cluster of the main factor (i.e., fuzzified value $M(1,1)$)'. Symbolically, it is expressed as:

Table 3 Different clusters of the main and the secondary factors using the *k*-means, *c*-means and the automatic clustering algorithm

Clusters generated by the *k*-means clustering algorithm [web error data set]							
Cluster no.	Clusters or intervals	Max	Min	Mid	Mean	Sum_deviation	Global_deviation
Clusters of the main factor							16.744
a_1	{3870, 2397, 3842, 3263, 3205, 4352, 2886, 3415, ...}	4352	2397	3374.5	3358	150.65	
a_2	{2493, 2593, 2644, 2657, 2701, 2894, 2819, 2787, 2910, 4845, 3169, 3455, ...}	4845	2493	3669			
a_3	{3626, 3087, 2435,...}	3626	2435	3030.5			
Clusters of the 1st secondary factor							
$a_{1,1}$	{0, 342, 0, 0, 0, 0, 0, 0, 0, 343, 1, 0,...}	343	0	171.5	167.83	3.623	
$a_{1,2}$	{0, 346, 342, 12, 0, 0, 0, 4,...}	346	0	173			
$a_{1,3}$	{0, 318,...}	318	0	159			
Clusters of the 2nd secondary factor							
$a_{2,1}$	{4, 10, 2, 6, 6, 3, 4, 6, 1, 9, 4, 6, 10,...}	10	1	5.5	7.33	0.7577	
$a_{2,2}$	{6, 3, 3, 6, 14, 10, 3, 4,...}	14	3	8.5			
$a_{2,3}$	{2,14,...}	14	2	8			
Clusters generated by the *c*-means web data set							
		Min	Max	Mid	Dev_diff		
Clusters of the main factor							
a_1	{3870, 2397, 3842, 3263, 3205, 4352, 2886, 3415, ...}	2397	4352	3374.5	3108.474474		
a_2	{3169, 2894, 2819, 2787, 2910, 3455, 2644, 4845, 2657, 2701, 2593, 3626, ...}	2593	4845	3719			
a_3	{3087, 2435,...}	3087	2435	2761			
Clusters generated by the Automatic Clustering Algorithm							
		Min	Max	Mid	Avg_diff	Dev_diff	
Clusters of the main factor							
a_1	{2397, 2435, 2493, 2593, 2644 2910, 3087, 3169, 3205, 3263, 3415, 3455, 3626, 3842, 3870, ...}	2046	4111	3078.5	97.45455	3108.474474	
a_2	{4352, 4845,...}	4111	4845	4478			
Clusters of the 1st secondary factor							
$b_{1.1}$	{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 4, 4, 12,...}	–82.96	165	41.02	15.72727	68.95013	
$b_{1.2}$	{318,342,342,243,346,...}	41.02	428.96	234.499			
Clusters of the 2nd secondary factor							
$b_{2.1}$	{1,...}	1	1	1	0.590909	138.6	
$b_{2.2}$	{2, 2,...}	2	2	2			
$b_{2.3}$	{3, 3, 3, 3,...}	3	3	3			
$b_{2.4}$	{4, 4, 4, 4,...}	4	4	4			
$b_{2.5}$	{6, 6, 6, 6, 6, 6,...}	6	6	6			
$b_{2.6}$	{9,...}	9	9	9			
$b_{2.7}$	{10, 10, 10,...}	10	10	10			
$b_{2.8}$	{14,14,...}	14	14	14			

$M(1, 1), S(1)_{1,1}, S(2)_{1,1} \rightarrow M(3, 1)$.

In case of the fuzzified one-step ahead occurrence of the main factor (i.e., error code 404) on 23/10/10, i.e., '#', a fuzzy logical relationship was established by applying Rule 2 (*Cf.* Sect. 3) of the developed algorithm as follows:

'If the 11th element of the 1st cluster of the main factor (i.e., fuzzified value $M(1, 11)$), the 17th element of the

Table 4 Fuzzy logical relationship among main (404), 1st secondary (406) and 2nd secondary (403) factors

Fuzzy logical relationship
$M(1,1), S(1)_{1,1}, S(2)_{1,1} \rightarrow M(3,1)$
$M(3,1), S(1)_{3,1}, S(2)_{1,1} \rightarrow M(1,2)$
$M(1,2), S(1)_{3,2}, S(2)_{1,2} \rightarrow M(1,3)$
$M(1,3), S(1)_{1,2}, S(2)_{3,3} \rightarrow M(1,4)$
$M(1,4), S(1)_{1,3}, S(2)_{2,4} \rightarrow M(1,5)$
$M(1,5), S(1)_{1,4}, S(2)_{1,5} \rightarrow M(3,2)$
$M(3,2), S(1)_{3,3}, S(2)_{2,1} \rightarrow M(2,10)$
$M(2,10), S(1)_{1,5}, S(2)_{1,2} \rightarrow M(2,4)$
$M(2,4), S(1)_{1,6}, S(2)_{3,3} \rightarrow M(2,5)$
$M(2,5), S(1)_{1,7}, S(2)_{3,4} \rightarrow M(2,2)$
$M(2,2), S(1)_{2,1}, S(2)_{2,5} \rightarrow M(3,3)$
$M(3,3), S(1)_{1,8}, S(2)_{2,2} \rightarrow M(1,6)$
$M(1,6), S(1)_{1,9}, S(2)_{1,6} \rightarrow M(2,11)$
$M(2,11), S(1)_{1,10}, S(2)_{1,6} \rightarrow M(2,6)$
$M(2,6), S(1)_{1,11}, S(2)_{3,7} \rightarrow M(2,7)$
$M(2,7), S(1)_{1,12}, S(2)_{1,8} \rightarrow M(2,8)$
$M(2,8), S(1)_{3,4}, S(2)_{2,9} \rightarrow M(2,9)$
$M(2,9), S(1)_{1,13}, S(2)_{1,10} \rightarrow M(1,7)$
$M(1,7), S(1)_{1,14}, S(2)_{2,7} \rightarrow M(1,8)$
$M(1,8), S(1)_{2,8}, S(2)_{1,7} \rightarrow M(1,9)$
$M(1,9), S(1)_{1,15}, S(2)_{1,8} \rightarrow M(2,3)$
$M(2,3), S(1)_{1,16}, S(2)_{3,11} \rightarrow M(2,1)$
…
$M(2,19), S(1)_{1,37}, S(2)_{2,32} \rightarrow \#$

1st cluster of the 1st secondary factor (i.e., fuzzified value $S(1)_{1,17}$), and the 12th element of the 2nd cluster of the 2nd secondary factor (i.e., fuzzified value $S(2)_{2,12}$) are at stage 21, then at stage 22 the fuzzified occurrence of the main factor will be '#'. Then, the fuzzy logical relationship would symbolically be expressed as:

$$M(1,11), S(1)_{1,17}, S(2)_{2,12} \rightarrow \#.$$

Different fuzzy logical relationships are contained in Table 4.

With the help of Rule 1 of the developed algorithm (*Cf.* Sect. 3), different known occurrences of the main factor given in Table 2 can be predicted. The defuzzified predicted value of M(1,5), i.e., 3,205, was calculated as follows (*Cf.* Sect. 3):

$$\text{predicted}(3636) = 3633$$

In the same way, the remaining known occurrences of the main factor can easily be predicted and shown in Table 5.

Again, using Rule 2 of the developed algorithm (*Cf.* Sect. 3), the occurrence of the main factor on 23/10/2011, i.e., #, can be predicted as follows:

$$\text{predicted}(\#) = \frac{3719 + 3374.5 + 2761 + 0}{3} \approx 3285.$$

Next, to check the predictive accuracy RMSE values are calculated as follows:

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(\text{Forecasted\_occurrence}_i - \text{Actual\_occurrence}_i)^2}{n}} \tag{20}$$

The variables, used in the above equation, are defined as follows:

Forecasted_occurrence$_i$: ith forecasted occurrence of the main factor.

Actual_occurrence$_i$ ith actual occurrence of the main factor.

Next, the outputs of the proposed algorithm were compared with that of the algorithm developed by Chen and Tanuwijaya (2011) using the automatic clustering algorithm and the results, given in Table 5, establish the superiority of the former. Afterwards, the outcomes of the proposed algorithm are compared with that of the algorithm developed by Chen and Tanuwijaya (2011), replacing the automatic clustering by k-means and c-means, respectively and the results are given in Table 5 which also shows the better predictive accuracy of the former. Quite interestingly, it is found that the predictive accuracy of the algorithm proposed by Chen and Tanuwijaya (2011) increases if the automatic clustering algorithm is replaced by the k-means and c-means, respectively as they can produce better quality clusters than the former (one possible reason). Additionally, it is also found that in this case, the quality of the clusters generated by the k-means algorithm (checking the DVI index, given in Table 1) is better than that of the c-means clustering approach and, as a consequence, better forecasted outputs are found from Chen and Tanuwijaya algorithm (2011) if the automatic clustering is replaced by k-means than c-means algorithm. The above discussion clearly establishes the influence of choosing suitable clustering algorithm on the forecasted results of the fuzzy time series-based prediction algorithms. The results are shown in Table 5. The RMSE, RMdSE and MdRAE values for the proposed forecasting algorithm is lesser than all of its competitors that can be found from the Tables 5, 9, 11, 12, 13 and 14. The bold portions of the tables confirms the propositions.

Moreover, the outputs of the proposed algorithm is compared with two statistical models, viz., MA(1) (univariate time series model) (Lutkepohl 2005) and VAR(1) (multivariate time series model) (Lutkepohl 2005) and found better predictive accuracy of the proposed algorithm. The corresponding MA(1) model is given as follows:

$$\text{MA}(1) : X_T = Z_T - 0.2024 Z_{T-1},$$

Table 5 Forecasted outcomes (approx.) along with theRMSE, RMdSE, MdRAE values for the post sample period

Original	Chen and Tanuwijaya			Proposed algorithm with k-means	MA(1)	VAR(1)
	Automatic	c-means	k-means			
3871	3681.1	3374.5	3419	3810	–	3870
3636	3790.29	3352.25	3794.5	3633	–	3021.972
2391	2379.9	2737	2579	2388	–	3079.081
3842	3800.6	3460.5	3301.5	3802	3270.07	3725.4
3203	3029	3170.75	3012	3100	3138.1	3400.52
3215	3099.5	3141.75	2983	3269	3138.1	3169.742
3047	3012.67	3082.75	3230	3046	3129.3	3175.025
4845	4762	4749.1	4109.75	4845	3147	3192.863
2697	2580	2865.25	3188	2680	3147	3182.917
2801	2890	2889.75	3210	2942	3138.1	3288.237
2993	2972	2835.75	3156	2889	3147	3281.256
2535	2887	4502.561	3077	2687	3129.3	3305.21
4252	4391	3118.75	3863.25	4100	3138.1	3693.958
3269	2781	2986.25	2965	3295	3138.1	2917.842
2994	3019	2948.75	3306.5	2959	3138.1	3175.37
2919	3231	2932.75	3269	2924	3129.3	3237.059
2887	2875	2994.25	3253	2972	3120.5	3255.083
2910	2617	2962.55	3314.5	2981	3129.3	3258.708
2986	3151	3246.75	3302.5	3054	3138.1	3612.5
3315	3011	3266.75	3088	3277	3120.5	3238.139
3555	3617	2861.25	3108	3533	3138.1	3125.881
2644	2998	2785.75	3181	2943	3138.1	3113.724
…	…	…	…	…	…	…
RMSE	867.1063	573.6783	437.5398	**32.90529**	595.4581	3200.03
RMdSE	815.1	513.7	417.5	**29.92**	515.8	3175.3
MdRAE	807.1	501.8	407.8	**27.79**	501.8	3112.3

where $\{Z_T\}$ is the white noise series corresponding to the series of the occurrences of 404 error code, i.e., $\{X_T\}$ (given in Table 2). The MA coefficients are determined with the help of the maximum likelihood method. Here, the white noise variance corresponding to $\{X_T\}$ can be calculated as 0.369892×10^6 whereas, the standard error of the MA coefficients can be calculated as 0.324468. The AICC and BIC (Lutkepohl 2005) of the proposed MA(1) model can be calculated as 0.364795×10^3 and 0.360366×10^3, respectively. The predicted occurrences of different web errors are given in Table 5. Similarly, the corresponding VAR(1) model is given as follows:

$$\begin{pmatrix} Y_T \\ X_{1T} \\ X_{2T} \end{pmatrix} = \begin{pmatrix} -0.2218 & 1.0948 & 0.6846 \\ -0.05292 & -0.01587 & -15.08921 \\ -0.002207 & -0.004290 & -0.019705 \end{pmatrix} \times \begin{pmatrix} Y_{T-1} \\ X_{1T-1} \\ X_{2T-1} \end{pmatrix} + \begin{pmatrix} 3892.9216 \\ 460.21221 \\ 14.357282 \end{pmatrix} + \begin{pmatrix} -16.7561 \\ -10.67506 \\ -0.080135 \end{pmatrix},$$

where Y_T = occurrences of error code 404 (main factor), X_{1T} = occurrences of the error code 406 (first secondary factor) and X_{2T} = the occurrences of the error code 403 (the second secondary factor). Again, $(3892.9216\ 460.21221\ 14.357282)^{\mathrm{T}}$ and $(-16.7561\ -\ 10.67506\ -\ 0.080135)^{\mathrm{T}}$ are the constant and the trend components of the above-mentioned VAR(1) model. The forecasted outputs of the VAR(1) model are shown in Table 5. Moreover, from Fig. 2, it is clear that the predicted accuracy of the proposed algorithm is better than all the other approaches used in the present study.

Additionally, χ^2-goodness of fit test too was carried out to validate the developed multivariate fuzzy forecasting algorithm. Here, $\chi^2_{\text{Computed}} = 20.83 < 40.289 = \chi^2_{\text{Tabulated}}$ at 22 degrees of freedom and 1 % level of significance for the data set given in Table 2. Therefore, the developed algorithm stands fully validated.

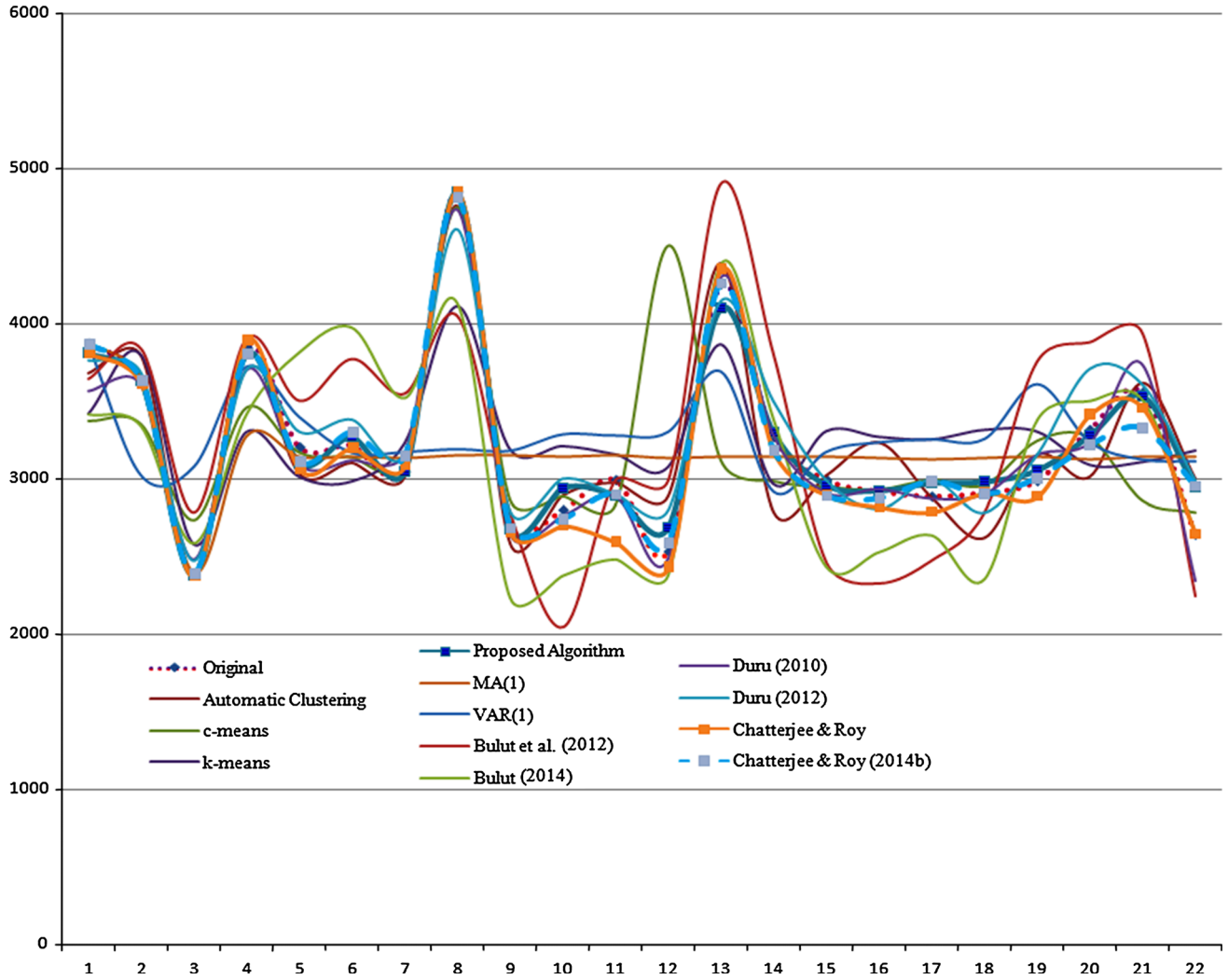

Fig. 2 Original and the predicted occurrences of 404

4.2 An example regarding coal processing

This sub section showcases a real example of the oil agglomeration (Sahinoglu and Uslu 2011) process for the beneficiation of coal fines (coal washing), where the environment is much different from that of the websites. Oil agglomeration can be used for separation of particles suspended in water differing in affinity towards oil drops. The affinity of particles suspended in water towards oil drops is called aquaoleophilicity (Sahinoglu and Uslu 2011). The term aquaoleophilicity reflects the fact that particles like oil drops in water. This property in similar to the hydrophobicity utilized in flotation in which the oil drop is substituted with the gas bubble. Successful oil agglomeration requires vigorous stirring to disperse oil drops and particles to facilitate sufficient number of collisions between them (Sahinoglu and Uslu 2011). For this purpose, impellers are required as the main equipment (Sahinoglu and Uslu 2011). Different properties of the impellers (independent variables) along with the experimental (dependent variable) and predicted % yield (clean coal) that used in this experiment are given in Tables 6 and 9. Figure 3 pictorially demonstrates the oil agglomeration process. Different abbreviations of the independent variables used in Tables 6 and 9 are given as follows:

$\mathbf{I_B}$: Number of impellers blades; $\mathbf{I_N}$: number of impellers; $\mathbf{I_D}$: diameters of the impellers; $\mathbf{I_W}$: width of the impellers' blades; **RPM:** impeller's speed (in revolution per minute).

The experimental yield (%) of the clean coal is the dependent variable. Here, the experimental yield (%) of the clean coal has been predicted based on I_B, I_N, I_D, I_W and RPM (independent variables). For the experimental purpose, in this paper a data set of length 81 has been used. Some part of the data set has been shown in Table 6. Here, for the modeling purpose 70 % of the data set, i.e., $(81 \times 0.7) \approx 57$, data have been used and the remaining are used for prediction purpose (post sample period).

From the first row of Table 6, it is found that if the number of impellers' blade is 2, number of impellers are 4, diam-

Table 6 Different independent and dependent variables of the oil agglomeration data set along with their respective positions in the clusters generated by the k-means clustering algorithm

Independent variables					Dependent
I_B	I_N	I_D	I_W	RPM	Experimental yield %
$2 = S(1)_{3,1}$	$4 = S(2)_{1,1}$	90	$20 = S(4)_{3,1}$	$1200 = S(5)_{2,1}$	$58.2300 = M(2, 1)$
$3 = S(1)_{1,1}$	$4 = S(2)_{1,2}$	90	$20 = S(4)_{3,2}$	$1200 = S(5)_{2,2}$	$63.1500 = M(3, 1)$
$4 = S(1)_{2,1}$	$4 = S(2)_{1,3}$	90	$20 = S(4)_{3,3}$	$1200 = S(5)_{2,3}$	$65.0000 = M(1, 1)$
$5 = S(1)_{2,2}$	$4 = S(2)_{1,4}$	90	$20 = S(4)_{3,4}$	$1200 = S(5)_{2,4}$	$65.2500 = M(1, 2)$
$4 = S(1)_{2,3}$	$1 = S(2)_{2,1}$	90	$20 = S(4)_{3,5}$	$1200 = S(5)_{2,5}$	$59.2900 = M(2, 2)$
$4 = S(1)_{2,4}$	$2 = S(2)_{2,2}$	90	$20 = S(4)_{3,6}$	$1200 = S(5)_{2,6}$	$62.5500 = M(3, 2)$
$4 = S(1)_{2,5}$	$3 = S(2)_{2,3}$	90	$20 = S(4)_{3,7}$	$1200 = S(5)_{2,7}$	$63.0000 = M(3, 3)$
$4 = S(1)_{2,6}$	$4 = S(2)_{1,5}$	90	$20 = S(4)_{3,8}$	$1200 = S(5)_{2,8}$	$65.0000 = M(1, 3)$
$4 = S(1)_{2,7}$	$5 = S(2)_{3,1}$	90	$20 = S(4)_{3,9}$	$1200 = S(5)_{2,9}$	$65.0000 = M(1, 4)$
$4 = S(1)_{2,8}$	$3 = S(2)_{2,4}$	75	$20 = S(4)_{3,10}$	$1200 = S(5)_{2,10}$	$59.5000 = M(2, 3)$
$4 = S(1)_{2,9}$	$3 = S(2)_{2,5}$	90	$20 = S(4)_{3,11}$	$1200 = S(5)_{2,11}$	$63.0000 = M(3, 4)$
$4 = S(1)_{2,10}$	$3 = S(2)_{2,6}$	100	$20 = S(4)_{3,12}$	$1200 = S(5)_{2,12}$	$63.2400 = M(3, 5)$
$4 = S(1)_{2,11}$	$3 = S(2)_{2,7}$	110	$20 = S(4)_{3,13}$	$1200 = S(5)_{2,13}$	$63.5600 = M(3, 6)$
$4 = S(1)_{2,12}$	$3 = S(2)_{2,8}$	100	$15 = S(4)_{2,1}$	$1200 = S(5)_{2,14}$	$57.3600 = M(2, 4)$
$4 = S(1)_{2,13}$	$3 = S(2)_{2,9}$	100	$20 = S(4)_{3,14}$	$1200 = S(5)_{2,15}$	$63.2400 = M(3, 7)$
$4 = S(1)_{2,14}$	$3 = S(2)_{2,10}$	100	$25 = S(4)_{1,1}$	$1200 = S(5)_{2,16}$	$66.7300 = M(1, 5)$
$4 = S(1)_{2,15}$	$3 = S(2)_{2,11}$	100	$20 = S(4)_{3,15}$	$800 = S(5)_{3,1}$	$55.2400 = M(2, 5)$
$4 = S(1)_{2,16}$	$3 = S(2)_{2,12}$	100	$20 = S(4)_{3,16}$	$1200 = S(5)_{2,17}$	$63.2400 = M(3, 8)$
$4 = S(1)_{2,17}$	$3 = S(2)_{2,13}$	100	$20 = S(4)_{3,17}$	$2400 = S(5)_{1,1}$	$64.2900 = M(1, 6)$
…	…	…	…	…	…

Dickey–Fuller test		
Series	p values	Results
I_B	$0.01 < 0.05$	Stationary
I_N	$0.01 < 0.05$	Stationary
I_D	$0.01 < 0.05$	Stationary
I_W	$0.01 < 0.05$	Stationary
RPM	$0.01 < 0.05$	Stationary
Experimental yield %	$0.01 < 0.05$	Stationary

eters of the impellers are 90 mm., width of the impeller blade is 20 mm., and the speed of the impellers are 1200 RPM, then the experimental yield (%) of the clean coal is 58.2300 %, whereas, the predicted yield (%) of clean coal by the developed algorithm and the ANN approach are 57.37 % and 62.17563 %, respectively (Table 9). The main motive behind citing this example is to unveil the extensive applicability of the developed prediction algorithm in different parts of science and technology. In this case, the ceil of the experimental yield (dependent variable), i.e., Experimental Yield, is considered as the main factor whereas, I_B, I_N, I_D, I_W and RPM (all the independent variables) are considered as the secondary factors. The following table shows different instances or observations of the independent and dependent variables involved in this experiment.

Applying step 1 and step 2 of the developed algorithm (*Cf.* Sect. 3) on the dataset (shown in Table 6) non- overlapping clusters generated by the k-means, automatic and c-means clustering algorithms are shown in Table 7. Next, to choose the suitable clustering algorithm, initially, the DVI of the clusters generated by the aforementioned clustering algorithm are calculated and are shown in Table 1, from which it is quite clear that the k-means clustering algorithm can produce better quality clusters. Consequently, in this case, the k-means clustering algorithm has been applied to partition the data set.

The *mean, sum_deviation* and the *global_deviation* (*Cf.* Sect. 3) are given as follows:

$$\text{mean} = 62.625, \text{sum\_deviation} = 2.165,$$
$$\text{global\_deviation} = 14.07.$$

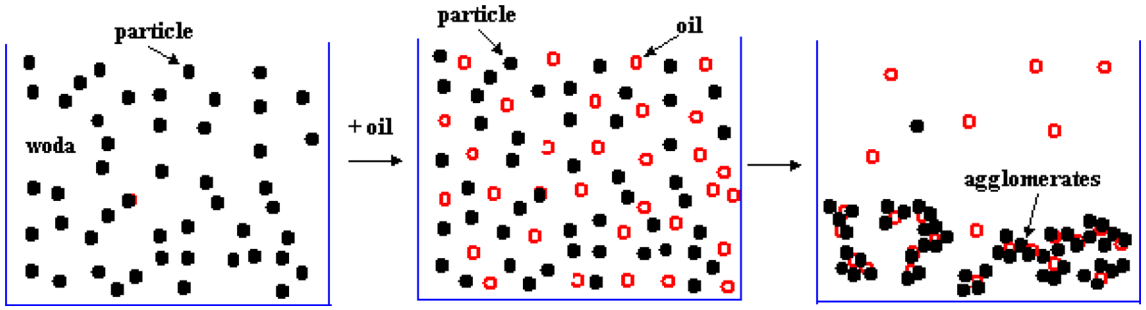

Fig. 3 Oil agglomeration process

Table 7 Clusters of the main and the secondary factors generated by the above-mentioned clustering algorithm

Clusters generated by the *k*-means clustering algorithm [oil data set]						
Cluster no.	Clusters or intervals	Min	Max	Mid	Mean	Sum_deviation
Clusters of the main factor						2.165
a_1	{65, 65.25, 66.73, 63.29,…}	65	66.73	65.865	62.625	
a_2	{63.15, 62.55, 63, 66.73, 64.29,…}	62.55	66.73	64.64		
a_3	{58.23, 59.29, 59.5, 57.36, 55.24,…}	55.24	59.5	57.37		
Clusters generated by the *c*-means clustering algorithm						
		Min	Max	Mid	Avg_diff	
Clusters of the main factor						
a_1	{58.23, 59.29, 59.5, 57.36, …}	57.36	59.5	58.43	0.63834	
a_2	{55.24,…}	55.24	55.24	27.62		
a_3	{63.15, 65, 65.25, 62.55, 63, 63.15, 65, 59.29, 62.55, 63.56, 63.24, 66.73, 63.24, 64.29, . . . }	62.55	66.73	64.64		
Clusters generated by the automatic cluetsring algorithm						
		Min	Max	Mid	Avg_diff	Dev_diff
Clusters of the main factor						
a_1	{55.24, 57.36, 58.23, 59.29, 59.5, 62.55, 63, 63, 63.15, 63.24, 63.24, 63.24, 63.56, 64.29, 65, 65.25, 66.73, . . . }	23.456	98.55	37.53	0.63834	63.56886
Clusters of the 1st secondary factor						
$b_{1.1}$	{2, 3, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 5, . . . }	0.074	6.928	6.93	0.16667	3.855269
Clusters of the 2nd secondary factor						
$b_{2.1}$	{1, 2, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 4, 4, 4, 4, 4, 5, . . . }	–0.5	5.5	5	0.23	3
Clusters of the 3rd secondary factor						
$b_{3.1}$	{75, 90, 90, 90, 90, 90, 90, 90, 90, 90, 90, 100, 100, 100, 100, 100, 100, 100, 100, . . . }	27.62	157. 39	92.5	1.94444	94.77869
Clusters of the 4rd secondary factor						
$b_{4.1}$	{15, 20, 20, 20, 20, 20, 20, 20, 20, 20, 20, 20, 20, 20, 20, 20, 20, 20, 25, . . . }	4.98	35.025	20	5.277778	20.04994
Clusters of the 5th secondary factor						
$b_{5.1}$	{800, 1200, 1200, 1200, 1200, 1200, 1200, 1200, 1200, 1200, 1200, 1200, 1200, 1200, 1200, 1200, 1200, 2400, . . . }	188.88	3011.13	1411.13	88.88889	1222.24752

In each case k-means clustering algorithm with three clusters was applied and as a consequence, the dataset corresponding to all the main and secondary factors were divided into 3 clusters each.

The fuzzy set representation for A_0 (the linguistic variable corresponding to a_0) is as follows [*Cf.* Eq. (7)]:

$$A_0 = \frac{1}{a_0} + \frac{\left(1 - \frac{1}{36}\right)}{a_1} + \frac{0}{a_3} + \sum_{j=1}^{5}\sum_{i=1}^{3} \frac{\left(1 - \frac{x_{j,i}}{n_{j,i} * n_p}\right)}{b_{j,i}} = \frac{1}{a_0} + \frac{0.97}{a_1} + \frac{0}{a_3}$$

Fuzzy set representation for the other linguistic variables for the main factor can be similarly defined. The fuzzy set representations for $B_{1,4}$ is given as follows:

$$B_{1,4} = \frac{0}{b_{1,0}} + \frac{0}{b_{1,1}} + \frac{0}{b_{1,2}} + \frac{0}{b_{1,3}} + \frac{1}{b_{1,4}} + \frac{0}{b_{1,5}} + \frac{0}{b_{1,6}} + \frac{0}{b_{1,7}} + \frac{0}{b_{1,8}} + \frac{0}{b_{1,9}}.$$

The fuzzy set representation for M(2,1) is defined as follows [*Cf.* Eq. (12)]:

$$\mathrm{M}(2,1) = \frac{1}{a_2} + \frac{0}{a_1} + \frac{0}{a_3} + \sum_{j=1}^{5}\sum_{i=1}^{3}\sum_{l} \frac{g_G(M(2,1)\_S(j)_{i,l})}{b_{j,i}}$$

Table 8 Fuzzy logical relationships of the data set given in Table 6

Fuzzy logical relationships
$M(2,1), S(1)_{3,1}, S(2)_{1,1}, S(3)_{2,1}, S(4)_{3,1}, S(5)_{2,1} \rightarrow M(3,1)$
$M(3,1), S(1)_{1,1}, S(2)_{1,2}, S(3)_{2,2}, S(4)_{3,2}, S(5)_{2,2} \rightarrow M(1,1)$
$M(1,1), S(1)_{2,1}, S(2)_{1,3}, S(3)_{2,3}, S(4)_{3,3}, S(5)_{2,3} \rightarrow M(1,2)$
$M(1,2), S(1)_{2,2}, S(2)_{1,4}, S(3)_{2,4}, S(4)_{3,4}, S(5)_{2,4} \rightarrow M(2,2)$
$M(2,2), S(1)_{2,3}, S(2)_{2,1}, S(3)_{2,5}, S(4)_{3,5}, S(5)_{2,5} \rightarrow M(3,2)$
$M(3,2), S(1)_{2,4}, S(2)_{2,2}, S(3)_{2,6}, S(4)_{3,6}, S(5)_{2,6} \rightarrow M(3,3)$
$M(3,3), S(1)_{2,5}, S(2)_{2,3}, S(3)_{2,7}, S(4)_{3,7}, S(5)_{2,7} \rightarrow M(1,3)$
$M(1,3), S(1)_{2,6}, S(2)_{1,5}, S(3)_{2,8}, S(4)_{3,8}, S(5)_{2,8} \rightarrow M(1,4)$
$M(1,4), S(1)_{2,7}, S(2)_{3,1}, S(3)_{2,9}, S(4)_{3,9}, S(5)_{2,9} \rightarrow M(2,3)$
$M(2,3), S(1)_{2,8}, S(2)_{2,4}, S(3)_{2,10}, S(4)_{3,10}, S(5)_{2,10} \rightarrow M(3,4)$
$M(3,4), S(1)_{2,9}, S(2)_{2,5}, S(3)_{2,11} S(4)_{3,11}, S(5)_{2,11} \rightarrow M(3,5)$
$M(3,5), S(1)_{2,10}, S(2)_{2,6}, S(3)_{1,1}, S(4)_{3,12}, S(5)_{2,12} \rightarrow M(3,6)$
$M(3,6), S(1)_{2,11}, S(2)_{2,7}, S(3)_{3,1}, S(4)_{3,13}, S(5)_{2,13} \rightarrow M(2,4)$
$M(2,4), S(1)_{2,12}, S(2)_{2,8}, S(3)_{1,2}, S(4)_{2,1}, S(5)_{2,14} \rightarrow M(3,7)$
$M(3,7), S(1)_{2,13}, S(2)_{2,9}, S(3)_{1,3}, S(4)_{3,14}, S(5)_{2,15} \rightarrow M(1,5)$
$M(1,5), S(1)_{2,14}, S(2)_{2,10}, S(3)_{1,4}, S(4)_{1,1}, S(5)_{2,16} \rightarrow M(2,5)$
$M(2,5), S(1)_{2,15}, S(2)_{2,11}, S(3)_{1,5}, S(4)_{3,15}, S(5)_{3,1} \rightarrow M(3,8)$
$M(3,8), S(1)_{2,16}, S(2)_{2,12}, S(3)_{1,6}, S(4)_{3,16}, S(5)_{2,17} \rightarrow M(1,6)$
...
$M(1,6), S(1)_{2,17}, S(2)_{2,13}, S(3)_{1,7}, S(4)_{3,17}, S(5)_{1,1} \rightarrow \#$

The defuzzified predicted value of M(3,1) can be calculated as follows [*Cf.* Eq. (13)]:

$$\text{predicted}(\mathrm{M}(3,1)) = \frac{1 * 64.64 + \left\{\left(1 - \frac{65-63.15}{2.165}\right) + \left(1 - \frac{65.25-63.15}{2.165}\right) + \left(1 - \frac{63.15-63}{2.165}\right)\right\} * 65.865}{1 + \left\{\left(1 - \frac{65-63.15}{2.165}\right) + \left(1 - \frac{65.25-63.15}{2.165}\right) + \left(1 - \frac{63.15-63}{2.165}\right)\right\}} \approx 65.26.$$

Different fuzzified occurrences of the main and the secondary factors as also the possible fuzzy relationships are tabulated and shown in Tables 6 and 8.

Again, from the first row of Table 16, using Rule I (*Cf.* Sect. 3), the following fuzzy logical relationship was formed:

'If the 1st element of the 2nd cluster of the main factor [i.e., fuzzified value M(2,1)], 1st element of the 3rd cluster of the 1st secondary factor [i.e., fuzzified value $S(1)_{(3,1)}$], 1st element of the 1st cluster of the 2nd secondary factor [i.e., fuzzified value $S(2)_{(1,1)}$], 1st element of the 2nd cluster of the 3rd secondary factor [i.e., fuzzified value $S(3)_{(2,1)}$], 1st element of the 3rd cluster of the 4th secondary factor [i.e., fuzzified value $S(4)_{(3,1)}$], and 1st element of the 2nd cluster of the 5th secondary factor [i.e., fuzzified value $S(5)_{(2,1)}$] are at stage 1, then at the next stage the fuzzified main factor will be M(3,1)'. The possible fuzzy logical relationships are tabulated and shown in Table 8 below.

Similarly, the other predicted experimental yields (%) can be calculated which are shown in Table 9.

Tables 9, 13 show the forecasted outputs of the proposed and several other prediction methods [proposed by Chen and Tanuwijaya 2011 (using automatic, c-means and k-means clustering algorithms), ANN, VAR(1), MA(3), Holt-Winter, Box-Jenkins, Bulut et al. 2012; Bulut 2014; Duru 2010, 2012; Chatterjee and Roy 2014a, b] with their RMSE, RMdSE and MdRAE values, which establishes the better efficiency and accuracy of the former. The pictorial representations of the above results are shown in Fig. 4. Comparing the DVI values it has been found that the quality of the clusters generated by the k-means clustering algorithm is better than the other two aforementioned clustering methods. Quite interestingly, it has been found that the forecasting accuracy of the Chen and Tanuwijaya (2011) method is enhanced if the automatic clustering technique is replaced by the k-means clustering algorithm. From the above discussion, it is quite

Table 9 Forecasted outputs (approx.) of different prediction methods for the oil agglomeration data set

Experimental yield (%)	Proposed with k-means	Chen and Tanuwijaya			ANN	VAR(1)	MA(3)	Holt-winter	Box-jenkins
		Automatic	k-means	c-means					
58.2300	57.37	37.53	57.38	58.325	62.17	37.13	–	59	69
63.1500	65.26	47.87	57.8	58.28	59.97	32.14	–	59	69
65.0000	64.98	51.336	63.57	63.9	63.65	34.92	–	67	77
65.2500	65.38	51.138	67	64.82	67.65	37.70	–	71	81
59.2900	57.37	51.38	57.13	64.95	62.47	36.12	–	71	81
62.5500	63.13	48.40	58.33	58.80	63.03	33.13	61.4	63	73
63.0000	64.16	50.039	63.28	63.6	59.67	35.97	66.1	67	77
65.0000	64.98	50.26	63.5	63.83	63.65	34.92	62.6	67	77
65.0000	64.98	51.26	57	64.82	63.65	37.76	62.9	71	81
59.5000	58.17	51.26	57	64.83	62.47	36.30	62.9	71	81
63.0000	64.75	48.51	58.44	58.92	59.67	34.02	62.9	63	73
63.2400	64.16	50.039	63.5	63.83	61.92	37.6	62.9	67	77
63.5600	64.71	50.38	63.62	63.94	59.97	41.35	62.9	67	77
57.3600	64.82	50.54	63.78	64.10	59.40	36.32	62.9	71	81
63.2400	64.75	47.444	57.37	57.96	61.97	37.69	62.9	59	79
66.7300	64.75	50.38	63.62	62.89	64.95	44.88	62.9	62	72
55.2400	64.82	52.079	63.78	63.94	58.15	37.93	62.9	67	77
63.2400	64.75	46.38	57.37	65.69	59.97	37.69	62.9	65	75
64.2900	65.00	50.38	63.6	40.24	60.65	37.13	62.9	47	67
…	…	…	…	…	…	…	…	…	…
RMSE	**2.3804**	14.3019	4.494	6.7538	2.7273	26.086	62.977	3.639	7.119
RMdSE	**2.134**	13.419	3.194	6.5131	2.1272	25.1097	61.107	3.169	7.669
MdRAE	**2.204**	13.268	3.187	6.5135	2.103	25.016	61.987	3.338	7.189

clear that the clustering technique has an influence on the fuzzy time series-based forecasting methods which corroborates well with the findings of Huarng (2001a). Hence, the proposed algorithm employs the k-means clustering technique to partition the data set given in Table 6 for better predictive accuracy. Next, the proposed forecasting algorithm can consider the contributions of different secondary factors on the defuzzified predicted occurrences of the main factor, which makes it more realistic than the other existing, extensively used fuzzy time series-based forecasting algorithms. Finally, the proposed algorithm has been compared with the ANN (Aladag et al. 2008) approach and two statistical methods, viz., VAR(1) (multivariate time series model), MA(3) (univariate time series model) to establish its better predictive accuracy. The corresponding VAR(1) model is given as follows:

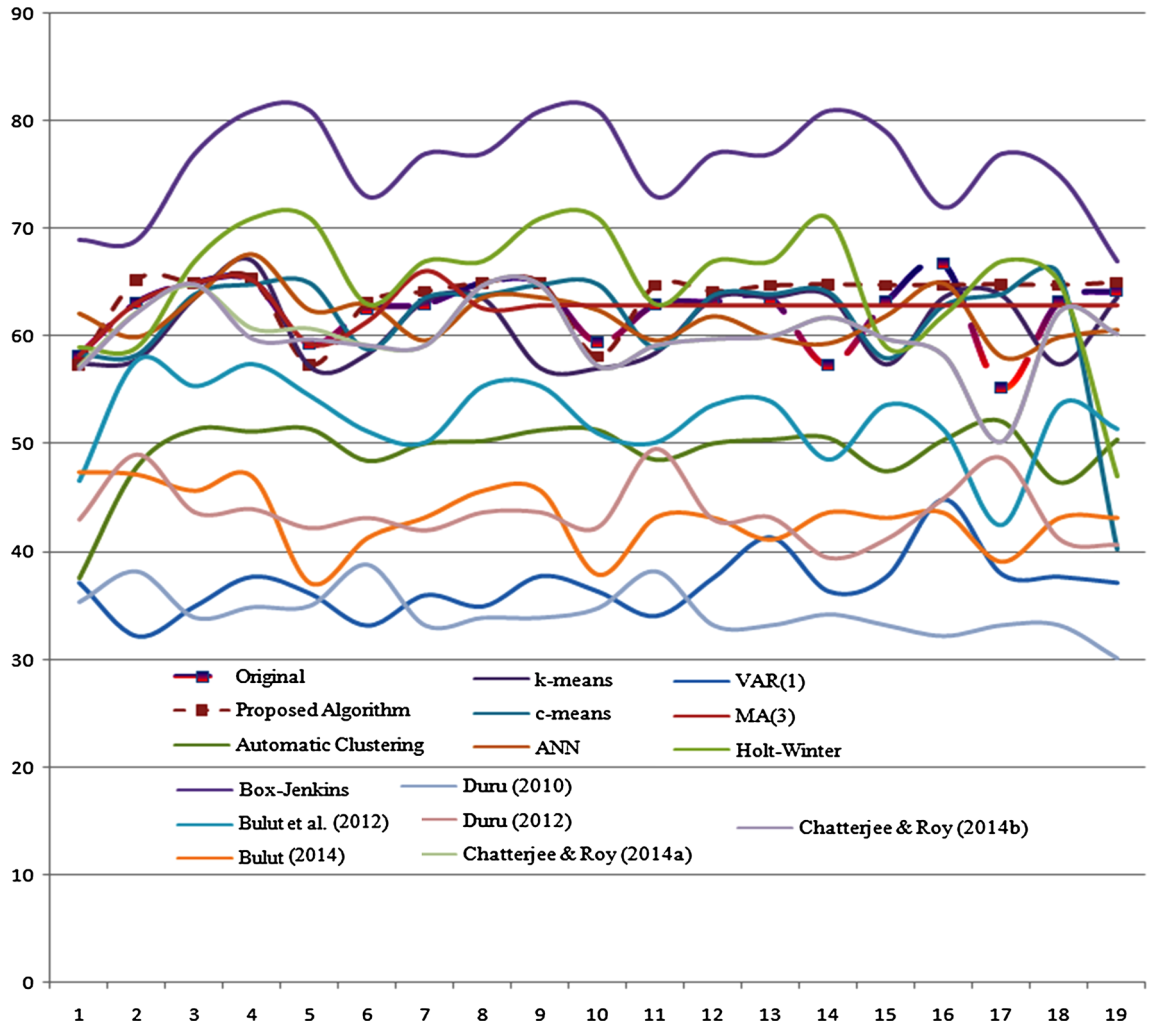

Fig. 4 The pictorial representation of the original and the predicted values (approx.) of the oil agglomeration data set

$$\begin{pmatrix} Y_t \\ X_{1t} \\ X_{2t} \\ X_{3t} \\ X_{4t} \\ X_{5t} \end{pmatrix} = \begin{pmatrix} 1.94318 & -4.72287 & -2.83968 & -0.36636 & -2.60480 & -0.03826 \\ 0.004213 & 0.016857 & 0.0467286 & 0.0041039 & 0.0004224 & -0.0002691 \\ 0.210601 & -1.749276 & 0.014056 & -0.033934 & -0.197276 & -0.003480 \\ 1.99407 & -2.80078 & -5.28686 & 0.14733 & -1.95601 & -0.03234 \\ 0.81219 & -1.03003 & -1.06770 & -0.19284 & -.77454 & -0.01384 \\ 173.799 & -220.979 & -226.111 & 24.399 & -205.855 & -2.551 \end{pmatrix} \begin{pmatrix} Y_{t-1} \\ X_{1t-1} \\ X_{2t-1} \\ X_{3t-1} \\ X_{4t-1} \\ X_{5t-1} \end{pmatrix} + \begin{pmatrix} 97.17546 \\ 3.7110951 \\ 7.627039 \\ 51.93884 \\ 23.98482 \\ 825.107 \end{pmatrix} + \begin{pmatrix} 0.27969 \\ -0.0210704 \\ 0.038676 \\ 0.80671 \\ 0.21784 \\ 47.180 \end{pmatrix},$$

where $(97.17546\ 3.7110951\ 7.627039\ 51.93884\ 23.98482\ 825.107)^{\mathrm{T}}$ and $(0.27969\ -0.0210704\ 0.038676\ 0.80671\ 0.21784\ 47.180)^{\mathrm{T}}$ are the constant and the trend components of the proposed VAR(1) model. Similarly, the MA(3) is given as follows:

$$\mathrm{MA}(3): X_t = Z_t - 0.5711Z_{t-1} - 0.5727Z_{t-2} + 0.9981Z_{t-3},$$

where $\{X_t\}$ is the series of the experimental yield (%) and $\{Z_t\}$ is the corresponding white noise. The white noise vari-

ance corresponding to $\{X_t\}$ can be calculated as 2.964867 whereas, the standard errors of the MA coefficients are calculated as 0.01408, 0.010068 and 0.01408. The corresponding AICC and BIC (Lutkepohl 2005) of the MA(3) model can be calculated as 91.686132 and 84.536399, respectively.

Additionally, the $\chi^2_{\text{computed}} = 1.688 < \chi^2_{\text{tabulated}} = 7.05$ at 99 % confidence level shows that the proposed method is fully validated in case of the present example. Figure 4 presents the pictorial representations of the original and the predicted values of different elements of the oil agglomeration data set, using different algorithms used in the present study and also shows that the predictive accuracy of the proposed algorithm is better than all of its competitors.

4.3 An example regarding finance data forecasting

In this sub section, the proposed algorithm is applied on a real financial data set, collected from the Ministry of Statistics and Program Implementation, Govt. of India (http://mospi.nic.in/Mospi_New/upload/asi/mospi_asi_rate_list.pdf), to show its efficiency, accuracy and applicability on financial data forecasting. The frequency of the data set is only 18, which is quite small for modelling as well as prediction. With this in mind, in this case the entire data set has been used for modeling as well as prediction purpose. The detailed description of the data set is given in the aforementioned web site. Here, the no. of records (year-wise) is considered to be the main factor, however, number of schedules, users in India and the users outside India are considered as the 1st, 2nd and 3rd secondary factors, respectively. For simplicity of calculation, the author has considered only three clusters for the k-means and the c-means clustering algorithms. Different clusters of the main and the secondary factors generated by the k-means, automatic clustering and the c-means clustering are tabulated and are shown in Table 10.

From Table 11 it can be found that in case of the finance data set the forecasted output by the Chen and Tanuwijaya method improves if its clustering technique (automatic clustering) is replaced by hard c-means and k-means clustering algorithm. Consequently, Table 11 shows that the RMSE of the Chen and Tanuwijaya method improves up to 8 times (approximately) in case of the k-means clustering algorithm, however, in case of the c-means clustering algorithm that is improved up to 3 times (approximately). The above discussion clearly evinces the influences of the selection of the suitable clustering algorithm on the forecasted output and corroborates well with the findings of Huarng and Yu (2006). Further, to check the quality of the clusters generated by the automatic clustering algorithm (2011), k-means clustering algorithm and the c-means clustering algorithm, shown in Table 10, the DVI indices are calculated which clearly shows that the k-means algorithm (among the aforementioned three) is the most suitable clustering technique for the finance data set (given in http://mospi.nic.in/Mospi_New/upload/asi/mospi_asi_rate_list.pdf). Hence, the above study strongly establishes that the poor selection of clustering strategy may hamper the forecasted outputs of the fuzzy time series based prediction algorithms. For example, from Table 11, it is clearly seen that the forecasted outcomes of the algorithm proposed by Chen and Tanuwijaya (2011) are improved if the automatic clustering algorithm is replaced by c-means and k-means clustering algorithms. To overcome this drawback, the proposed algorithm provides the flexibility to choose suitable clustering technique and in turn the RMSE decreases. In this case, the proposed algorithm employs the k-means clustering algorithm for the clustering purpose due to its suitability of the data set.

The existing extensively used fuzzy time series-based forecasting algorithms do not incorporate the influences of the secondary factors at the time of defuzzification of the main factor, which is, however, removed by the proposed algorithm. Consequently, Table 11 shows the superiority of the proposed algorithm in terms of RMSE over the algorithm proposed by Chen and Tanuwijaya (2011) (using the automatic clustering method, k-means and c-means), MA(3) and the VAR(1) models. The corresponding MA(1) model is given as follows:

$$x(t) = z(t) + 0.2996\, z(t-1).$$

Additionally, the $\chi^2_{\text{computed}} = 24.98 < \chi^2_{\text{tabulated}} = 25.989$ at 90 % confidence level shows that the proposed method is fully validated in case of the present example. The proposed algorithm has been compared with the algorithms proposed by Chen and Tanuwijaya (2011) (using automatic, c-means and k-means clustering algorithms), MA(1), Holt-Winter, Box-Jenkins, (Bulut et al. 2012; Bulut 2014; Duru 2010, 2012; Chatterjee and Roy 2014a, b) and the corresponding results are given in Table 11. The pictorial representation of the above results is shown in Fig. 5. Figure 5 shows the better predictive accuracy of the proposed algorithm.

4.4 Comparison with the 'traditional four step' algorithm

The membership values of the 'traditional four step' algorithms (Aladag et al. 2008; Bulut et al. 2012; Bulut 2014; Chen et al. 2013; Chen and Tanuwijaya 2011; Chen 1996; Duru 2010, 2012; Duru and Bulut 2014; Dunn 1973; Huarng 2001a, b; Huarng and Yu 2005, 2006; Mamdani 1977; Ross 2010; Song and Chissom 1993a, b, 1994; Tanaka 1996; Tseng et al. 2001; Zadeh 1975) can only be 0, 0.5 and 1. On the other hand, the membership values in case of the proposed algorithm can be more real numbers lying in the interval [0,1] apart from 0, 0.5 and 1. It is to be remembered that most of the existing 'traditional four step' algorithms can only be used in case of static length intervals. Quite on the contrary, the developed algorithm is itself capable to take

Table 10 Different clusters of the finance data generated by the aforementioned clustering algorithms

Clusters generated by the *k*- means clustering algorithm [financial data]	
Clusters of the main factor	
a_1	{543115, 1544154, 1599949}
a_2	{1526444, 1634492}
a_3	{161391, 1047407, 1093914, 1408012, 1674753, 1700939, 1733773, 1742098, 1759856, 1810409, 1981445, 1992578}
Clusters of the 1st secondary factor	
$b_{1,1}$	{28723, 41846, 42242, 49340, 54348, 56557, 56888}
$b_{1,2}$	{(25332, 33515, 40059, 41096, 56889, 57304, 57771, 58617, 59825)}
$b_{1,3}$	{57926}
Clusters of the 2nd secondary factor	
$b_{2,1}$	{30159, 43939, 44354, 51807, 57065, 59385, 59732}
$b_{2,2}$	{(26599, 35191, 42062, 43151, 59733, 60169, 60660, 61548, 62817)}
$b_{2,3}$	{60823}
Clusters of the 3rd secondary factor	
$b_{3,1}$	{177890, 259304, 261718, 305685, 336757, 350436, 352458}
$b_{3,2}$	{156969, 207600, 248204, 254600, 352458, 355037,358008, 363166, 370635}
$b_{3,3}$	{358854}
Clusters by the Chen and Tanuwijaya algorithm	
a_1	{161391, 543115, 1047407, 1093914, 1408012, 152444, 1544154, 1599949, 1634492, 1674753, 1700939, 1733773, 1742098, 1759856, 1810409, 1981445, 1992578, 2081116}
$b_{1,1}$	{25332, 28723, 33515, 40059, 41096, 41846, 42242, 49340, 54348, 56557, 56888, 56889, 57304, 57771, 57926, 58617, 59825, 66875}
$b_{2,1}$	{26599, 30159, 35191, 42062, 43151, 43939, 44354, 51807, 57065, 59385, 59732, 59733, 60169, 60660, 60823, 61548, 62817, 70219}
$b_{3,1}$	{156969, 177890, 207600, 248204, 254600, 259304, 261718, 305685, 336757, 350436, 352458, 352458, 355037, 358008, 358854, 363166, 370635, 414313},
Clusters of the main factor generated by the *c*-means clustering algorithm	
a_1	{1733773, 1093914}
a_2	{1599949, 543115, 1047407}
a_3	{161391, 1408012, 152444, 1544154, 1634492, 1674753, 1700939, 1742098, 1759856, 1810409, 1981445, 1992578, 2081116}

care of both static and variable-sized overlapping as well as non-overlapping intervals. The effects of the previous and the very next elements of a particular point can only be considered in case of almost all the existing 'traditional four step' algorithms. In case of the developed algorithm, the effects of all the elements, present in the data set can be considered for predicting a particular element. This feature makes the developed algorithm more flexible and also superior to the 'traditional four step' algorithms. Apart from this, the developed algorithm can take care of both stationary as well as non-stationary data sets, which cannot be found in case of other 'traditional four step' algorithms. Consequently, the predictive accuracy of the proposed algorithm increases.

From the foregoing analysis and discussion of the algorithm implementation test results, it is safely concluded that the developed algorithm is not only accurate but is also superior to the other existing algorithms.

4.5 Comparison with some other well-known, recently developed fuzzy time series-based algorithm

This sub section presents a comparative study of the proposed algorithm with some well-known recently developed fuzzy time series-based forecasting algorithms as follows:

There is a huge number of recently developed well-known fuzzy time series-based forecasting algorithms available in literature (Bulut 2014; Bulut et al. 2012; Duru 2010, 2012; Chatterjee and Roy 2014a, b). The work of Duru (2010) is suffered from the equi-spaced and fixed sized intervals. Moreover, for partitioning purpose Duru (2010) has developed his own clustering strategy that may not be able to generate best quality clusters from different types of data sets. Hence, the predictive accuracy of the algorithm has been affected as the lengths of the intervals have an influence on the predicted accuracy of the proposed algorithm (Huarng

Table 11 Original and different predicted outcomes (approx.) of the finance data along with their RMSE, RMdSE and MdRAE

Original	Proposed with k-means	Chen and Tanuwijaya			MA(1)	Holt–Winter algorithm	Box–Jenkins
		Automatic clustering	k-means	c-means			
1700939	1701989	5166866.6	1076984.5	1121253.5	–	5112546.6	5119901
1599949	1599147	5095792.1	1386235.5	1411096.25	–	5054112.1	5054989
1408012	1408182	4999823.6	1338466.75	1335740.5	–	4912473.6	4912998
1634492	1632792	5113063.6	1494365	1264632.75	1956700	5115433.6	5115970
1674753	1674853	5133193.1	1607650	1377872.75	1989700	5170113.1	5170978
1733773	1733879	5362704.1	1375868.75	1398003.25	2104400	5311014.1	5311981
543115	543189	4567375.1	1402652.5	1573808.25	2104400	4510015.1	4510976
1047407	1047079	4819521.1	810049.75	8073235	2104400	4816091.1	4816991
1093914	1093956	4827774.6	1062195.75	1059469.5	2104400	4812094.6	4812959
161391	171452	4376513.1	1085449.25	1253878.75	2104400	4317223.1	4317991
1526444	1526474	5059039.6	616461.5	641322.25	2104400	5011409.6	5011989
1544154	1544284	5166866.6	1298988	1323848.75	2104400	5111046.6	5111946
2081116	2081617	5336375.6	1867008	1332703.75	2104400	5323115.6	5323999
1810409	1810439	5201022.1	1579050.25	1601184.75	2104400	5200022.1	5200899
1981445	1988444	5286540.1	1443696.75	1465831.25	2704400	5212000.1	5212999
1992578	1992558	5292106.5	1529214.75	1551349.25	2804400	5212026.5	5212926
1759856	1759835	5175745.6	1534781.25	1556915.75	2504400	5130265.6	5130999
1742098	1742088	5166866.6	1418420.25	1440554.75	2504400	5110676.6	5110986
RMSE	**633.4374**	3569949.56	464814.43	1737853.6	950450.7	1664420.18	1664958.96
RMdSE	**603.49**	3569949.56	464814.43	1737853.6	950450.7	1664420.18	1664958.96
MdRAE	**599.79**	3569949.56	464814.43	1737853.6	950450.7	1664420.18	1664958.96

Dickey–Fuller test for stationarity checking

Series	p value	Result
Original	0.01 <0.05	Stationary
1st secondary factor	0.01 <0.05	Stationary
2nd secondary factor	0.01 <0.05	Stationary

2001a). Consequently, the proposed algorithm becomes data-dependent. Again, at the time of forecasting the main factors, the contributions of the secondary factors are not considered. Moreover, the membership values are only 0, 0.5 and 1. Again, this algorithm is not able to judge the stationarity of the data set, i.e., if the data set is non-stationary then also the prediction mechanism remains the same, which can be considered as a major drawback. Latter, Duru (2012) developed a fuzzy integrated logical forecasting model for dry bulk shipping index forecasting, in which, again the membership values have been taken only 0, 0.5 and 1. Moreover, the sizes of the intervals have been considered as fixed and the stationarity of the data set has not been checked.

In their extensive study, Bulut et al. (2012) have developed a fuzzy integrated logical forecasting (FILF) model of time charter rates in dry bulk shipping, which is mainly a vector autoregressive design of fuzzy time series with fuzzy c-means clustering algorithm. But this approach is again data-dependent as the fuzzy c-means clustering algorithm may not be able to partition all the data sets into best quality clusters. It can be verified with the help of corresponding DVI values (Dunn 1973). Again, in the latter year, Bulut (2014) has modified his previous approach by modeling seasonality using the fuzzy integrated logical forecasting (FILF) approach, which is, however, not free from all the aforementioned drawbacks.

In some recent studies, Chatterjee and Roy (2014a, b) have developed two novel fuzzy time series-based forecasting algorithms, which are, however, not free from certain important drawbacks in the modeling techniques. The first major drawback in the modeling technique of these algorithms is the inability to judge whether the data set is stationary or non-stationary. The non-stationary behaviors can be trends, cycles, random walks or combinations of the

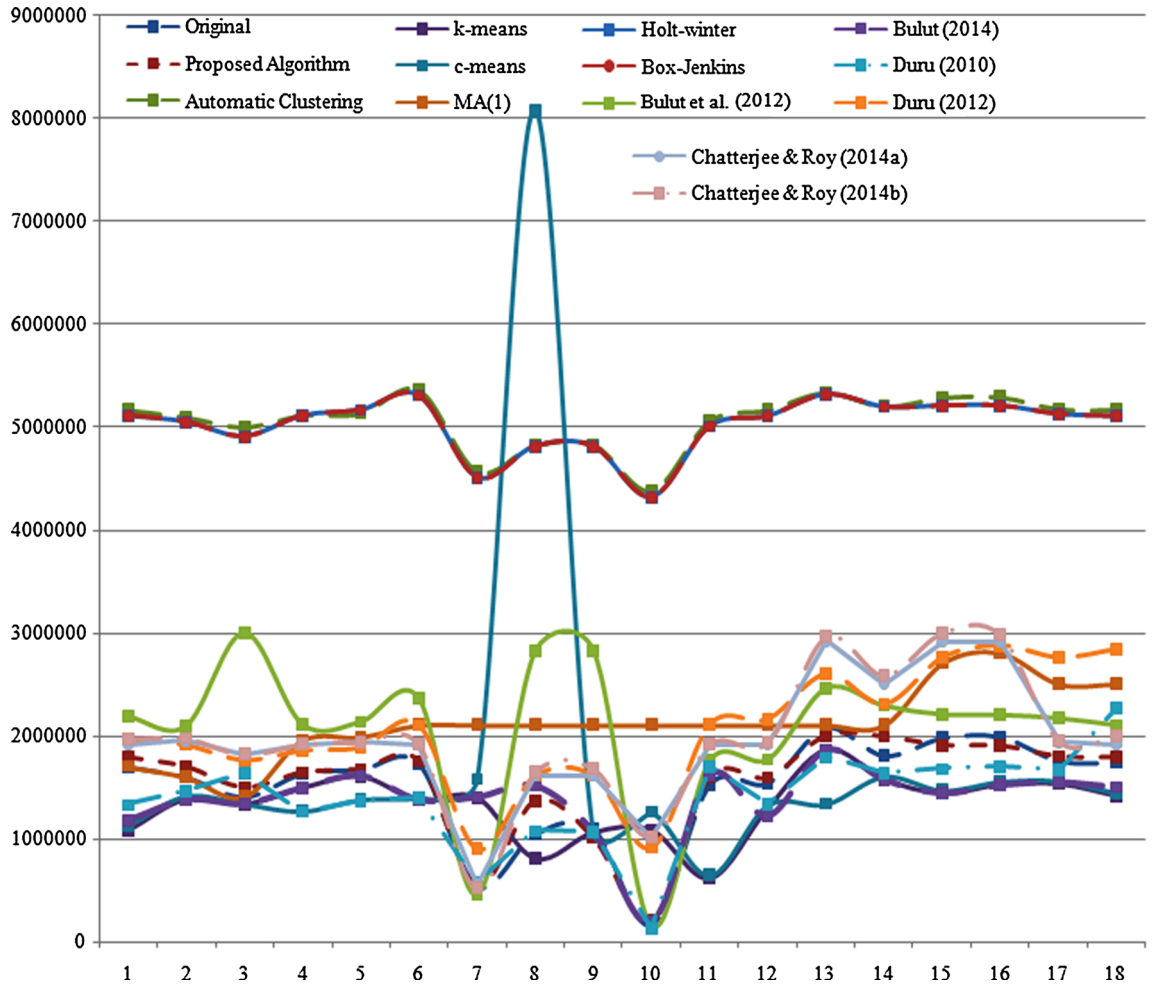

Fig. 5 The pictorial representation of the original and the predicted values (approx.) of the finance data set

three. Non-stationary data, as a rule, are unpredictable and cannot be modeled or forecasted (Lutkepohl 2005). The results obtained using non-stationary time series may be spurious in that they may indicate a relationship between two variables where one does not exist (Lutkepohl 2005). As a consequence, the predictive accuracy of the algorithms (Chatterjee and Roy 2014a, b) reduces. Another major drawback in the aforementioned algorithms is that in both the cases the authors (Chatterjee and Roy 2014a, b) have developed their own clustering algorithms, which may not be able to generate best quality clusters for all types data sets (clearly shown in Table 1) and, as a consequence, the predictive accuracy of the forecasting algorithms decreases (Huarng 2001a, b). Moreover, developing own clustering algorithms, in turn, makes the corresponding fuzzy time series-based forecasting algorithms data-dependent, as all the clustering algorithms are not suitable for all type of data sets. This is because of the fact that all the clustering algorithms are not able to generate best quality clusters of all the data sets. With this in mind, in the present paper, at first the suitable clustering algorithm for the data set has been chosen and then the proposed forecasting algorithm has been applied. Again, the algorithm proposed by Chatterjee and Roy (2014b) does not incorporate the influences of the secondary factors on the main factor at the time of defuzzification, which can be considered as a severe drawback in the modeling technique. However, this drawback has been removed in the proposed algorithm.

Additionally, the algorithms proposed by Chatterjee and Roy (2014a, b) have presented both sample and post-sample period results to investigate both estimation accuracy and forecasting accuracy, respectively. However, any developed forecasting method must confirm that the post-sample period is not used for clustering the data set. If the aforementioned algorithms (Bulut 2014; Bulut et al. 2012; Duru 2010, 2012; Chatterjee and Roy 2014a, b) are suitable for forecasting the unknown future, clusters should not be estimated by using test period since it has been assumed that they are unknown future values and, as a consequence, they may not contribute to the business practice. The pro-

Table 12 The occurrences of different frequently occurred error codes from 30/9/2010 to 22/10/2011 along with their positions in the respective clusters generated by the *k*-means clustering algorithm

Original main factor (404)	Proposed algorithm	Bulut et al. (2012)	Bulut (2014)	Duru (2010)	Duru (2012)	Chatterjee and Roy (2014a)	Chatterjee and Roy (2014b)
3871	3811	3641	3411	3566	3761	3809	3870
3636	3633	3833	3333	3603	3679	3605	3633
2391	2381	2781	2581	2481	2472	2376	2388
3842	3802	3902	3412	3702	3702	3891	3802
3203	3100	3500	3810	3090	3301	3057	3110
3215	3269	3769	3969	3119	3379	3196	3299
3047	3046	3546	3516	3166	3157	3086	3146
4845	4845	4045	4125	4735	4605	4845	4815
2697	2680	2780	2230	2645	2790	2656	2681
2801	2942	2042	2372	2763	2999	2700	2742
2993	2889	2989	2478	2889	2899	2592	2899
2535	2687	2987	2377	2488	2797	2434	2587
4252	4100	4900	4380	4300	4144	4351	4260
3269	3295	3795	3375	3275	3495	3168	3185
2994	2959	2459	2429	2909	2987	2893	2890
2919	2924	2324	2524	2933	2804	2818	2874
2887	2972	2472	2632	2872	2992	2786	2982
2910	2981	2781	2351	2902	2781	2908	2901
2986	3054	3754	3374	3154	3154	2885	3004
3315	3277	3877	3497	3233	3707	3414	3217
3433	3533	3933	3503	3733	3603	3454	3324
2644	2943	2243	2653	2343	2987	2643	2954
3860	3810	3110	3530	3710	3610	3892	3771
…	…	…	…	…	…	…	…
RMSE	**32.91**	103.79	111.57	107.91	95.99	79.91	78.39
RMdSE	**29.92**	105.78	112.78	109.78	98.97	79.995	79.12
MdRAE	**27.79**	101.75	109.77	107.44	95.37	75.46	78.19

The corresponding RMSE, RMdSE and MdRAE values are also given in this Table

posed forecasting method can remove the above-mentioned drawbacks, making it a very powerful tool for forecasting. Apart from this, the proposed algorithm has better predictive accuracy than the aforementioned all the fuzzy time series-based forecasting algorithms. It can be easily found from Table 12, 13 and 14 as in each case the proposed algorithm has the least RMSE, RMdSE and MdRAE values (Hyndman 2006). From the above study, it is quite clear that the proposed algorithm is not only capable of removing all the drawbacks of the existing fuzzy time series-based forecasting algorithms, but also, can correctly incorporate the influences of different secondary factors on the main factor. As a result, the predictive accuracy of the proposed algorithm increases and the modeling becomes more realistic.

From Tables 12, 13 and 14 it can be found that the predictive accuracy of the proposed algorithm is highest and that of the algorithm developed by Chatterjee and Roy (2014a) remains in the second position for all the data sets used in the present study.

Apart from this, some more differences (regarding the modeling technique) between the proposed algorithm and the algorithms proposed by Chatterjee and Roy (2014a, b) have been given as follows:

(i) The function defined in the prediction Rule 1 [Eq. (10)] of the proposed algorithm is more realistic than that of Chatterjee and Roy (2014a, b). This is because of the fact that, in the former case the absolute distance between the two points (main, secondary factors) has been considered. Accordingly, the rules of predictions have been modified for the main as well as the secondary factors. However, in the latter cases the number of points which have distances greater than *sum_deviation* is considered. Consequently, the predictive accuracy of the proposed model increases. Moreover, the

Table 13 The original and the predicted values (approx.) of the oil agglomeration data set using the proposed and different fuzzy time series based approaches

Main factor	Proposed algorithm	Bulut et al. (2012)	Bulut (2014)	Duru (2010)	Duru (2012)	Chatterjee and Roy (2014a)	Chatterjee and Roy (2014b)
58.2300	57.37	46.57	47.41	35.35	42.97	57.23	56.93
63.1500	65.26	57.77	47.18	38.18	49.07	62.28	56.18
65.0000	64.98	55.376	45.68	33.9	43.65	64.72	59.82
65.2500	65.38	57.428	47	34.87	43.95	60.71	59.75
59.2900	57.37	54.47	37.13	34.99	42.17	60.71	59.71
62.5500	63.13	51.16	41.33	38.81	43.13	59.14	59.14
63.0000	64.16	50.139	43.21	33.16	41.97	59.14	59.14
65.0000	64.98	55.376	45.68	33.88	43.65	64.72	59.72
65.0000	64.98	55.376	45.68	33.88	43.65	64.72	59.72
59.5000	58.17	50.96	3790	34.78	42.27	57.23	57.23
63.0000	64.75	50.139	43.21	38.19	49.61	59.14	59.14
63.2400	64.16	53.639	43.15	33.18	42.97	59.74	59.74
63.5600	64.71	53.938	41.16	33.19	43.17	59.99	59.99
57.3600	64.82	48.54	43.68	34.19	39.41	61.73	59.73
63.2400	64.75	53.639	43.15	33.18	41.17	59.74	59.74
66.7300	64.75	51.28	43.61	32.19	44.99	58.18	58.18
55.2400	64.82	42.45	39.11	33.19	48.75	50.19	50.19
63.2400	64.75	53.639	43.15	33.18	41.17	62.28	59.28
64.2900	65.00	51.39	43.16	30.12	40.61	60.19	59.19
…	…	…	…	…	…	…	…
RMSE	**2.3804**	18.317	24.446	36.789	32.773	6.096	7.907
RMdSE	**2.134**	18.019	24.047	36.508	32.239	6.071	7.018
MdRAE	**2.204**	19.398	25.576	37.758	33.870	7.092	8.512

The corresponding RMdSE and MdRAE values are also given in this Table

algorithm developed by Chatterjee and Roy (2014b) is not able to consider the influences of different secondary factors at the time of defuzzification of the main factor.

(ii) In case of the algorithm proposed by Chatterjee and Roy (2014a) the *accuracy_factor* has to be chosen only based on the expert judgment as no hard and fast rule has been given by them. Hence, if the accuracy_factor $\in (0, \left|\frac{\text{mean\_distance}}{2}\right|] \subset \mathbb{R}$ is perfectly chosen, the predictive accuracy increases, otherwise, it will decrease. But, every time it is not possible to choose correct *accuracy_factor* and, as a consequence, the result deteriorates. Keeping this in mind, in the present paper, the author has removed this concept. For example, the *mean_distance* (Chatterjee and Roy 2014a) of the web error data set, given in Table 2 is 792. Hence, by Chatterjee and Roy (2014a)

$$0 < \text{accuracy\_factor} \leq \left|\frac{\text{mean\_distance}}{2}\right|;$$

$$\text{i.e.,} 0 < \text{accuracy\_factor} \leq \left|\frac{792}{2}\right|;$$

$$\text{i.e.,} 0 < \text{accuracy\_factor} \leq 396.$$

Hence, accuracy_factor$\in (0, 396] \subset \mathbb{R}$, i.e., *accuracy_factor* can be any real number among the infinitely many real numbers lying between 0 and 396, which is, however, one of the most difficult tasks. Hence, *accuracy_factor* choosing is the biggest challenge in case of the algorithm proposed by Chatterjee and Roy (2014a). Keeping this in mind, this concept has been removed from the present forecasting algorithm.

(iii) In the modern fast and competitive world every algorithm needs both accuracy and lesser computational complexity simultaneously, i.e., faster execution. This is because of the fact that modern people or different industries will choose the algorithm having better accuracy and lesser execution time. Hence, the proposed algorithm has also been compared in the ground of computational complexity (Knuth 1973) with the algorithms proposed by Chatterjee and Roy (2014a, b).

The computational complexity of the algorithm proposed by Chatterjee and Roy (2014a) can be calculated as $(C * \theta(s) + D * M(s))$; $C, D \in \mathbb{Z}^+$, except the calculation of the *accuracy_factor*. If it is calculated, the complexity

Table 14 The original and the predicted values (approx.) of the finance data set using the proposed and different fuzzy time series-based approaches

Main factor	Proposed algorithm k-means	Bulut et al. (2012)	Bulut (2014)	Duru (2010)	Duru (2012)	Chatterjee and Roy (2014a)	Chatterjee and Roy (2014b)
1700939	1701989	2186867	1176183	1326253	1966510	1914746	1969746
1599949	1599147	2095792	1376132	1461716	1906760	1954492	1973432
1408012	1408182	2999823	1338466	1636760	1759801	1814951	1818360
1634492	1632792	2113063	1494365	1261632	1852750	1915451	1914521
1674753	1674853	2133193	1607650	1367672	1882614	1945153	1937073
1733773	1733879	2362704	1375868	1396063	2102360	1914854	1914260
543115	543189	459375	1402652	573807	894400	548195	519595
1047407	1047079	2819521	1510949	1073735	1604260	1614891	1642691
1093914	1093956	2827774	1062495	1059779	1604940	1614394	1677794
161391	171452	157713	185049	127878	904941	1015023	1018443
1526444	1526474	1759039	1616361	1691922	2104380	1915109	1914389
1544154	1544284	1769866	1218688	1329130	2159800	1915346	1934646
2081116	2081617	2460375	1857018	1785713	2598400	2925215	2959815
1810409	1810439	2291012	1571050	1635174	2304598	2505212	2585712
1981445	1988444	2206540	1440696	1675130	2759800	2915200	2998200
1992578	1992558	2202196	1519214	1697589	2875900	2912026	2975926
1759856	1759835	2170149	1544781	1668705	2759400	1935495	1947895
1742098	1742088	2101876	1498420	2269051	2840400	1913954	1998454
RMSE	**633.4374**	167644.694	76779.51	56182.21	130383.071	114856.09	122087.14
RMdSE	**603.49**	160894.7	76081.6	56018.3	130314.1	114109.1	121014.1
MdRAE	**599.79**	160801.7	76076.6	56010.3	130301.1	114001.1	121007.1

The corresponding RMdSE and MdRAE values are also given in this table

increases heavily. This is because of the fact that every time an *accuracy_factor* has to be selected from the set $(0, \left|\frac{\text{mean\_distance}}{2}\right|]$ (having infinite number of elements, as open set has infinite elements) and the same value has been used for prediction. Continuing this process infinite number of *accuracy_factors* can be found along with infinite number of predicted values. The *accuracy_factor* corresponding to the best predicted data (having least RMSE, RMdSE and MdRAE) can be selected as the *accuracy_factor*. It may involve infinite number of comparisons. Hence, the above procedure increases the complexity of the algorithm greatly. Quite on the contrary, the computational complexity of the proposed algorithm (if k-means clustering algorithm has been adopted) is found to be best in partitioning the experimental data set. For this purpose, the DVI values (Dunn 1973) of the generated clusters can be compared) is at most $(O(nkdi) + D * M(s)); n, k, d, i, D \in \mathbb{Z}^+$, where $M(s)$ can be considered as the complexity of the chosen multiplication algorithm, when the inputs are two s-digit numbers. Again, n is the number of d-dimensional vectors, k the number of clusters and i the number of iterations needed until convergence (Knuth 1973). This will change if a new clustering algorithm has been adopted. But, still the complexity is less than the algorithm proposed by Chatterjee and Roy (2014a) as no clustering algorithm involves infinite number of comparisons.

On the other hand, the computational complexity of the algorithm proposed by Chatterjee and Roy (2014b) is $(C * \boldsymbol{M(s)} * \boldsymbol{M(s)} * \theta(s) + D * M(s)); C, D \in \mathbb{Z}^+$, which is greater than the proposed algorithm. This increment is because of the adopted clustering algorithm implementing the concept of Mahalanobis distance (Chatterjee and Roy 2014b). Hence, the computational complexity of the proposed algorithm is less than the algorithm proposed by Chatterjee and Roy (2014b). However, the complexity of the algorithms proposed by Chatterjee and Roy (2014a, b) is less than the algorithm proposed by Chen and Tanuwijaya (2011).

(iv) The proposed algorithm is easier to implement than the algorithms proposed by Chatterjee and Roy (2014a, b).

4.6 Analysis of the residuals

This subsection showcases the analysis of the residuals of the proposed multivariate fuzzy forecasting algorithm. The residual of an observed value is the difference between the observed value and the estimated function value. The main

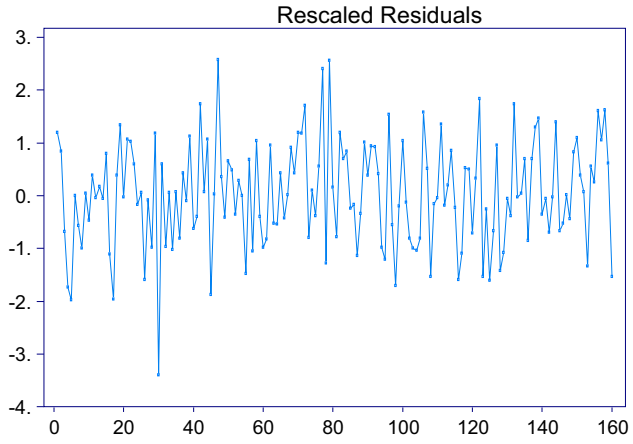

Fig. 6 The rescaled residuals of the web error data set

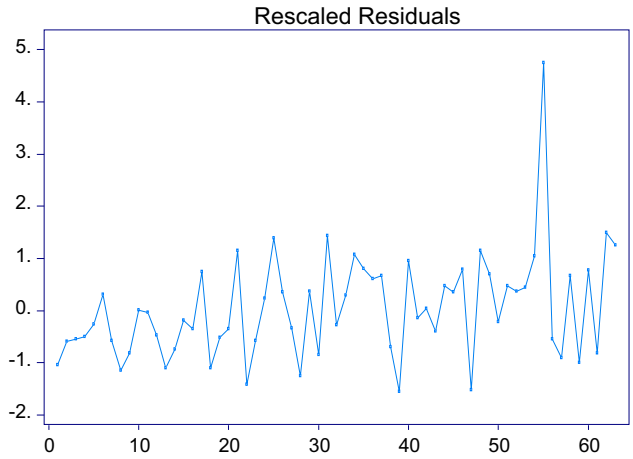

Fig. 7 The rescaled residuals of the oil agglomeration data set

motive of this analysis is to confirm that the remaining residuals do not follow any specific pattern and as a result they can be considered as the white noise (Lutkepohl 2005). Figure 6 confirms that the remaining residuals of the web error data set for the proposed algorithm do not include any pattern and, as a consequence, they can be considered as white noise. Similarly, Fig. 7 confirms that the remaining residuals of the oil agglomeration data set (given in Table 13) do not follow any pattern and hence, it can be considered as the white noise. In a similar manner, from Fig. 8, it can be found that the residuals of the finance data set for the proposed clustering algorithm can also be considered as the white noise as they do not follows any pattern.

5 Conclusion

The present paper demonstrates a novel multivariate fuzzy time series-based forecasting algorithm that is able to remove the drawbacks of the previously developed fuzzy time series-

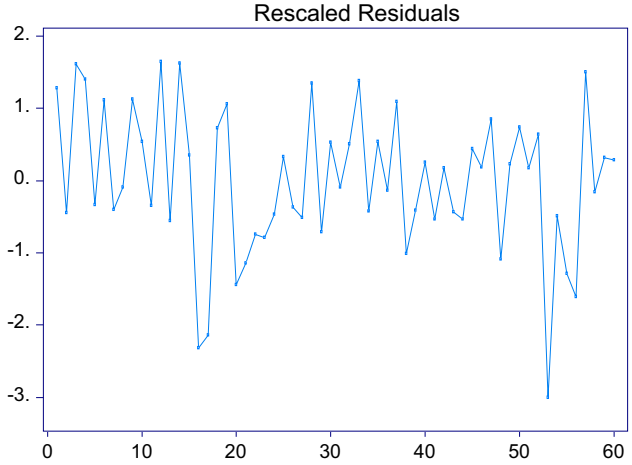

Fig. 8 The rescaled residuals of the finance data set

based techniques. Initially, the proposed algorithm can check the stationarity of the data set. If the data set is stationary, the proposed algorithm continues its different steps. Otherwise, it removes the non-stationarity of the data set and continues with the different steps of the proposed forecasting algorithm. Again, this novel algorithm can generate variable-sized clusters or intervals by applying a suitable clustering algorithm, assign more real numbers lying between 0 and 1 as the membership values of different elements, incorporate the effects of different secondary factors in the defuzzification process, which are, however, considered as the important findings arising out of this work. Moreover, the developed algorithm shows better predictive accuracy. For testing purpose, the developed algorithm was applied on three different domains, viz., oil agglomeration process for the beneficiation of the coal fines (coal washing technique), the frequently occurred web error prediction (a burning topic related to web technology) and financial data forecasting which manifests its applicability over broad domains like, the coal industries, web technology as well as finance. The real dataset related to the oil agglomeration for the beneficiation of coal fines was collected from CIMFER, Dhanbad, India (aCSIRLab, run by the Govt. of India), and that regarding the frequently occurred web error codes of www.ismdhanbad.ac.in, the official website of ISM Dhanbad, was collected from the Indian School of Mines Dhanbad, India server. However, the remaining data set was collected from the Ministry of Statistical and Program Implementation, Govt. of India. The proposed forecasting method was compared with thirteen different conventional (univariate and multivariate), e.g., VAR, MA, Holt–Winter, Box–Jenkins (Lutkepohl 2005), and fuzzy time series-based forecasting algorithms, viz., Bulut et al. (2012), Bulut (2014), Duru (2010, 2012), Chatterjee and Roy (2014a, b), Chen and Tanuwijaya (2011) (replacing its clustering algorithm with c-means and k-means techniques, respectively). Moreover, the

accuracy of the proposed algorithm has also been compared with the ANN approach (Aladag et al. 2008). But in every case, the proposed algorithm proves its efficiency and better predictive accuracy. Hence, from the above study it is quite clear that the proposed algorithm can be applicable over a large domain more accurately for forecasting purpose.

Acknowledgments The author is thankful to Dr. Henry Lieberman, Massachusetts Institute of Technology, Cambridge, Massachusetts,USA, for his thorough checking and wonderful suggestions for the betterment of this paper. The author is also thankful to Mr. Rajesh Mishra, system administrator, ISM Dhanbad, India, for providing the log files of www.ismdhanbad.ac.in Last, the author shows his gratitude to Dr. V. K. Kalyani, Scientist, CIMFER, a CSIR Lab (run by the Govt. of India), for providing the data related to the oil agglomeration process. The author is very much thankful to the reviewers for their valuable suggestions.

References

Aladag H, Basaran MA, Egrioglu E, Yolcu U, Uslu VR (2008) Forecasting in high order fuzzy time series by using neural networks to define fuzzy relations. Expert Syst Appl 3:4228–4231

Aliev RA, Fazlollahi B, Aliev RR, Guirimov B (2008) Linguistic time series forecasting using fuzzy recurrent neural network. Soft Comput 12:183–190

Bezdek JC, Ethrlich R, Full W (1984) FCM: the fuzzy c-means clustering algorithm. Comput Geosci 10:191–203

Bulut E, Duru O, Yoshida S (2012) A fuzzy integrated logical forecasting (FILF) model of time charter rates in dry bulk shipping: a vector autoregressive design of fuzzy time series with fuzzy C-means clustering. Marit Econ Logist 14(3):300–318

Bulut E (2014) Modeling seasonality using the fuzzy integrated logical forecasting (FILF) approach. Expert Syst Appl 41(4):1806–1812

Chatterjee S, Roy A (2014a) Web software fault prediction under fuzzy environment using MODULO-M multivariate fuzzy clustering algorithm and newly proposed revised prediction algorithm. Appl Soft Comput 22:372–396

Chatterjee S, Roy A (2014b) Novel algorithms for web software fault prediction. Qual Reliab Eng Int. doi:10.1002/qre.1687

Chen SM (1996) Forecasting enrollments based on fuzzy time series. Fuzzy Sets Syst 81(3):311–319

Chen SM, Tanuwijaya K (2011) Multivariate fuzzy forecasting based on fuzzy time series and automatic clustering techniques. Expert Syst Appl 38:10594–10605

Chen LS, Hsueh CC, Chang CJ (2013) A two-stage approach for formulating fuzzy regression models. Knowl Based Syst 52:302–310

Dunn JC (1973) A fuzzy relative of the ISODATA process and its use in detecting compact well- separated clusters. J Cybern 3(3):32–57

Duru O (2010) Fuzzy integrated logical forecasting model for dry bulk shipping index forecasting: an improved fuzzy time series approach. Expert Syst Appl 37:5372–5380

Duru O (2012) A multivariate model of fuzzy integrated logical forecasting method (M-FILF) and multiplicative time series clustering: A model of time-varying volatility for dry cargo freight market. Expert Syst Appl 39(4):4135–4142

Duru O, Bulut E (2014) A non-linear clustering method for fuzzy time series: histogram damping partition under the optimized cluster paradox. Appl Soft Comput. doi:10.1016/j.asoc.2014.08.038

Hartingan JA, Wong MA (1979) A K-means clustering algorithm. J Royal Stat Soc Ser C 28(1):100–108

Huarng K (2001b) Heuristic models of fuzzy time series for forecasting. Fuzzy Sets Syst 123:369–386

Huarng K (2001a) Effective lengths of intervals to improve forecasting fuzzy time series. Fuzzy Sets Syst 123:387–394

Huarng K, Yu THK (2005) A type 2 fuzzy time series model for stock index forecasting. Phys A 353:445–462

Huarng K, Yu THK (2006) Ratio-based lengths of intervals to improve fuzzy time series forecasting. IEEE Trans Syst Man Cybern Part B Cybern 32:328–340

Huynh T, Miller J (2009) Another viewpoint on evaluating web software reliability based on workload and failure data extracted from server logs. Empir Softw Eng 14:371–396

Hyndman RJ (2006) Another look at forecast-accuracy metrics for intermittent demand. Foresight Int J Appl Forecast 4:43–46

Khashei M, Bijari M, Hejazi SR (2012) Combining seasonal ARIMA models with computational intelligence techniques for time series forecasting. Soft Comput 16:1091–1105

Knuth DE (1973) The art of computer programing, vol 1. Addison-Wesley Publishing Company, USA

Lutkepohl H (2005) New introduction to multiple time series analysis. Springer, Berlin

Mamdani, E. H. (1977) Application of fuzzy logic to approximate reasoning using linguistic synthesis. IEEE Trans. On ComputersC-26, 1182–1191

Rabiei MR, Arghami NR, Taheri M (2014) Least-squares approach to regression modeling in full interval-valued fuzzy environment. Soft Comput 18(10):2043–2059

Ross TJ (2010) Fuzzy logic with engineering Applications, Wiley, India

Sahinoglu E, Uslu T (2011) Increasing coal quality by oil agglomeration after ultrasonic treatment. Fuel Process Technol 116:332–338

Shen J, Chang Lee ES, Deng Y, Brown SJ (2005) Determination of cluster number in clustering microarray data. Appl Math Comput 169(2):1172–1185

Song Q, Chissom BS (1993b) Fuzzy time series and its model. Fuzzy Sets Syst 54:269–277

Song Q, Chissom BS (1993a) Forecasting enrollments with fuzzy time series-Part I. Fuzzy Sets Syst 54:1–9

Song Q, Chissom BS (1994) Forecasting enrollments with fuzzy time series-Part II. Fuzzy Sets Syst 62:1–8

Tanaka K (1996) An introduction to fuzzy logic for practical applications, Springer

Tsaur RC (2008) Forecasting analysis by using fuzzy grey regression model for solving limited time series data. Soft Comput 12:1105–1113

Tseng FM, Tzeng GH, Yu HC, Yaun B (2001) Fuzzy ARIMA model for forecasting the foreign exchange market. Fuzzy Sets Syst 118:9–19

Zadeh LA (1975) The concept of linguistic variable and its application to approximate reasoning, parts 1–3. Inform Sci 8(3):199–249, 301–357, 9, 43–80