Analogy-Based Software Effort Estimation Using Fuzzy Numbers

Mohammad Azzeh, Daniel Neagu, Peter I. Cowling
AI Research Group, Department of Computing, University of Bradford
Bradford, U.K., BD7 1DP
M.Y.A.Azzeh@bradford.ac.uk, D.Neagu@bradford.ac.uk, P.I.Cowling@bradford.ac.uk

Abstract.

Context: Software effort estimation at early stage is a crucial task for project bedding and feasibility study. Since collected data at early stage of software development lifecycle is always imprecise and uncertain, it is very hard to deliver accurate estimate. Analogy-based estimation, which is one of the popular estimation methods, is rarely used at early stage because of uncertainty associated with attribute measurement and data availability.

Objective: in order to improve performance of analogy-based estimation at early stage, using all available early data, we integrated it with Fuzzy numbers. Particularly, this paper proposes a new software project similarity measure and a new adaptation technique based on Fuzzy numbers in order to support analogy-based estimation at early stage of software development lifecycle.

Method: Empirical evaluation with Jack-knifing procedure carried out, using five benchmark data sets of software projects, namely, ISBSG, Desharnais, Kemerer, Albrecht and COCOMO, are reported. The results are compared to those obtained by methods employed in the literature using case-based reasoning and stepwise regression.

Results: In all data sets the empirical evaluations have shown that the proposed similarity measure and adaptation techniques method were able to remarkably improve the performance of analogy-based estimation at early stage of software development. The results have also shown the proposed method outperforms some well know estimation techniques such as case-based reasoning and stepwise regression.

Conclusions: It is concluded that the proposed estimation model could form a useful approach for early stage estimation especially when data is almost uncertain.

Keywords: early stage software effort estimation, cost estimation, estimation by analogy, similarity measurement, generalized Fuzzy numbers.

1. Introduction

Early stage software estimation is very important for project bedding and feasibility study. However, this estimate is yet kind of guesses, even good predictions are not sufficient with inherent uncertainty and risks. Boehm (2006) reported that the uncertainty level of effort estimation at early stage of software development is very high, while this uncertainty range presents a decreasing trend as software development process goes towards last stages. Kitchenham et al. (1997) have investigated likely sources of software estimate uncertainty such as: (1) assumption error, (2) measurement error, (3) model error and (4) skill of estimator (Jorgensen, 2004), which interestingly confirms that software practitioners are aware of the irrefutable uncertainty in estimation and therefore expect some inaccuracy in the prediction output.

Despite the fact that analogy based estimation is one of the popular used means of making a prediction (Shepperd & Schofield, 1997; Auer & Biffl, 2004), it is rarely used at early stage of software development because of such inherent uncertainty and imprecision associated with attribute measurement at early stage. Attribute measurement is generally a kind of estimation that is equally likely to be above or below real value, therefore two projects that may seem similar may indeed be different in a critical way. Thus, the uncertainty in assessing similarities means that two different estimators could develop significantly different views and effort estimates.

To improve the performance of analogy-based estimation at early stage, we combined it with Fuzzy numbers in that new similarity measure and adaptation technique based on Fuzzy numbers are proposed and used in analogy based estimation. Fuzzy numbers were introduced by (Jain & Dubois, 1978; Jain, 1976, 1978), and are used in many research fields such as: control engineering (Wei & Chen, 2009a), statistics (Fuzzy regression, expected values) (Chen & Chen, 2003), and risk analysis (Chen & Chen, 2003; Lee, 1999). A Fuzzy number (Chen & Chen, 2003; Hsieh & Chen, 1999; Wei & Chen, 2009a) is a Fuzzy set that respects normal and convex properties of Fuzzy membership function. A more recent concept for Fuzzy number is the generalized Fuzzy number which is described in more details in section 3. In this paper, we propose a new similarity measure between two generalized Fuzzy numbers. It combines the concepts of geometric distance, the centre of gravity of generalized Fuzzy number and height of generalized Fuzzy numbers. We also prove three properties of the proposed similarity measure. Later on, we employed Fuzzy number concepts to develop a new adaptation technique. The proposed similarity measure and adaptation technique have been used to develop a new analogy software effort estimation model called "Generalized Fuzzy Number Software Estimation" (GFNSE) model. To the best of our knowledge, the concept of generalized Fuzzy number has not been used in software effort estimation by analogy, and this paper shows that it may offer a promising direction for research.

The rest of this paper is organized as follows: section 2 reviews literature on software effort estimation. Section 3 introduces the fundamental concepts of generalized Fuzzy numbers. Section 4 presents the proposed similarity measure between two generalized Fuzzy numbers. Section 5 illustrates the software prediction model based on generalized Fuzzy numbers. Section 6 introduces the evaluation criteria. Section 7 presents design of experiments and results of empirical evaluation. Section 8 presents discussion of our results. Finally, the paper ends with conclusions and future direction.

2. Related Work

Use Case Points and Function Points models are the most widely used estimation techniques at early stage. Even that, they have some challenges as they are environment dependent models because they need calibration, in addition to the uncertainty and incompleteness of the data set used (Pfleeger et al., 2005). These models are highly dependent on size input which needs reliable measurement (Idri et al., 2001; Azzeh et al., 2008a, 2008b). Moreover, experienced software estimators are required to translate the set of requirements into their likely number of use cases, actors and scenarios (Pfleeger et al., 2005). On the other hand, machine learning based estimation techniques such as analogy-based estimation and neural networks are rarely used at early-stage of software development because of the uncertainty associated with attribute measurement at early phases, and the need for accurate data for training and validation purposes.

Uncertainty at early stage is a pervasive problem in software effort estimation as the available data is almost imprecise and vague. Generally, uncertainty in software measurement is associated with a lack of precise knowledge such that software developers sometimes have measurements that are inaccurate, inexact, or of low confidence. For example, "Software testing might need 10 man-months", which exhibits a degree of imprecision (Kitchenham, 1997). Moreover, there is no estimation model can include all the factors that factually affect the effort required to accomplish software project (Jorgensen, 2004).

Song et al. (2005) proposed an early stage software effort estimation method based on Grey Relational Analysis (GRA) called GRACE. They employed GRA to select an optimal attribute set based on the similarity degree between dependent variable and other variables. The variables that exhibit large similarity are selected to form the optimal attribute set. The variables are preferably continuous rather than categorical. The GRA is later used to derive new estimate by finding the closest case that approximately agrees with current case on all effort drivers. Recently, Azzeh et al. (2009) developed a new similarity measure based on integration of Fuzzy set theory and Grey Relational Analysis for Analogy-based estimation. The proposed has capability to deal with numerical and categorical attributes in that two levels of similarity measure have been defined: local and global measures. The results obtained suggested that the proposed model produces good accuracy when compared to other well Known estimation techniques such as case-based reasoning, stepwise regression and artificial neural network. Idri et al. (2001) proposed a new Fuzzy analogy software cost estimation based on linguistic quantifiers. The model was designed for the datasets that are described by linguistic quantifiers (using an ordinal scale) such as the COCOMO dataset. They used Fuzzy aggregation operators to adjust estimates based on Fuzzy similarity between two software projects. This approach does not appear to perform well over other datasets that are not structurally similar to COCOMO dataset, and it is no suitable for early stage estimation. Musflek et al. (2000) developed a granular model for software cost estimation based on Fuzzy number called *f-COCOMO*. Both input (kilo line of code) and output (effort) are represented by their corresponding triangular Fuzzy numbers. The mapping between input and output was performed using the possibility distribution which assumes that the uncertainty in input domain should be reflected on the uncertainty in the output domain. The model has lack of validation in terms of prediction accuracy.

Above all, the rational of using Fuzzy numbers in EA stems from the vagueness presented in the exact values of some attributes used in the problem of effort estimation. This uncertain information has great impact on the similarity assessment between software projects and on the driving a new estimate. To deal with this kind of uncertain data, Fuzzy numbers provide a powerful approach to model and solve some of the challenges in EA.

3. Generalized Fuzzy Numbers

In this section, we briefly review some fundamental concepts of generalized Fuzzy numbers (Chen & Chen, 2008; Wei & Chen, 2009a, 2009b). A generalized trapezoidal Fuzzy number *A* is a Fuzzy subset of the real line *R* and represented as $A=[a,b,c,d;w]$, where $0<w\leq 1$ represents opinion confidence of estimator or decision maker (Chen & Chen, 2003). The elements *a*, *b*, *c* and *d* are real numbers. The membership of this Fuzzy number μ_A should satisfy the following conditions:

(1) μ_A is a continuous mapping from *R* to the closed interval in [0,1].

(2) $\mu_A(x)$=0, where $-\infty < x \leq a$. and $d \leq x \leq \infty$

(3) $\mu_A(x)$ is monotonically increasing in [a, b].

(4) $\mu_A(x)=w$ where $b \leq x \leq c$.

(5) $\mu_A(x)$ is monotonically decreasing in [c, d].

Although the common membership function of Fuzzy number is trapezoidal, it could be represented also by either triangular or Gaussian membership function. In this paper trapezoidal and triangular functions are the special interest in our study and we will leave Gaussian function for future work. Strictly speaking, in case of *w*=1, then the generalized Fuzzy number is regarded as normal trapezoidal Fuzzy number denoted as [*a, b, c, d*]. If *a=b* and *c=d* we obtain a crisp interval, and if *a<b=c<d* then *A* is represented as triangular Fuzzy number. In case of a*=b =c=d*, *A* is simply a real number. Later we will use $A=[a,b,c,d;w]$ to represent both generalized triangular and trapezoidal Fuzzy numbers with restriction for triangular function when *b* equals *c*.

Figure 1 shows example of two different generalized trapezoidal Fuzzy numbers $A=\left[0.2,0.4,0.5,0.7;\ 1\right]$ and $B=\left[0.2,0.4,0.5,0.7;\ 0.8\right]$. Similarly, Figure 2 shows two different generalized triangular Fuzzy numbers $A=\left[0.2,0.4,0.4,0.7;\ 1\right]$ and $B=\left[0.2,0.4,0.4,0.7;\ 0.8\right]$.

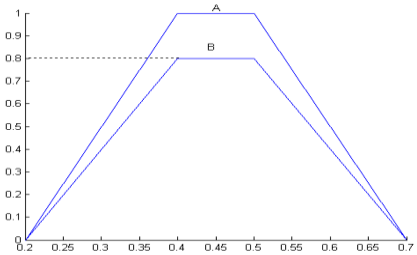

Figure 1. Two generalized trapezoidal Fuzzy numbers: A=[0.2, 0.4, 0.5, 0.7; 1] and B=[0.2, 0.4, 0.5, 0.7; 0.8]

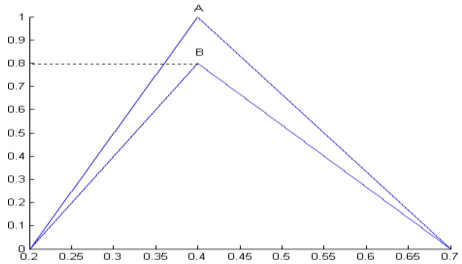

Figure 2. Two generalized triangular Fuzzy numbers: A= [0.2, 0.4, 0.4, 0.7; 1] and B=[0.2, 0.4, 0.4, 0.7; 0.8]

Chen (1985) proposed some arithmetic operations for generalized Fuzzy numbers that have been later subject of debate by Hsieh et al. (1999) who emphasized that these arithmetic operations do not change the shape of generalized Fuzzy number after carrying out any arithmetic operation. The Fuzzy arithmetic operations proposed by Chen (1985) are introduced below:

Let $A=\left[a_1,a_2,a_3,a_4;w_A\right]$, $B=\left[b_1,b_2,b_3,b_4;w_B\right]$, $w_A,w_B\in\left[0,1\right]$

(1) Generalized Fuzzy numbers addition ($\oplus$):

$$A\oplus B=(a_1+b_1,a_2+b_2,a_3+b_3,a_4+b_4;\min(w_A,w_B)) \qquad (1)$$

where a_1, b_1, c_1, d_1, a_2, b_2, c_2 and d_2 are any real numbers

(2) Generalized Fuzzy numbers subtraction (Θ):

$$A\ \Theta\ B=(a_1-b_1,a_2-b_2,a_3-b_3,a_4-b_4;\min(w_A,w_B)) \qquad (2)$$

where a_1, b_1, c_1, d_1, a_2, b_2, c_2 and d_2 are any real numbers

(3) Generalized Fuzzy numbers multiplication ($\otimes$):

$$A\otimes B=(a_1\times b_1\ ,\ a_2\times b_2\quad a_3\times b_3,\ a_4\times b_{41}\ ,\min(w_A,w_B)) \qquad (3)$$

where a_1, b_1, c_1, d_1, a_2, b_2, c_2 and d_2 are positive real numbers

(4) Generalized Fuzzy numbers division (Ø)

$$A \, Ø \, B = (a_1 / b_1, a_2 / b_2, a_3 / b_3, a_4 / b_4; \min(w_A, w_B)) \quad (4)$$

where a_1, b_1, c_1, d_1, a_2, b_2, c_2 and d_2 are all nonzero positive real numbers

4. The proposed similarity measure

In this section, we propose a similarity measure which aims to capture the vagueness presented in attribute values that are used in the early stage software estimation process. Strictly speaking, we present a new method to calculate degree of similarity between two generalized Fuzzy numbers. The proposed method combines the concepts of geometric distance, centre of gravity (COG) and height of generalized Fuzzy number. The similarity degree between two generalized Fuzzy numbers is affected by many facts such as: area of Fuzzy number, COG of Fuzzy number, shape of the Fuzzy number, position of Fuzzy number and height of Fuzzy number (Wei & Chen, 2009a, 2009b).

Suppose there are two generalized Fuzzy numbers $A = [a_1, a_2, a_3, a_4; w_A]$ and $B = [b_1, b_2, b_3, b_4; w_B]$, where $0 \le a_1 \le a_2 \le a_3 \le a_4 \le 1$, $0 \le b_1 \le b_2 \le b_3 \le b_4 \le 1$ and $w_A, w_B \in [0,1]$ represent the height of generalized Fuzzy numbers A and B. The degree of similarity $S(A, B)$ between two generalized Fuzzy numbers is a composition of three elements, namely, Height Adjustment Ratio (HAR), Geometric Distance (GD), and Shape Adjustment Factor (SAF) as shown in Eq. (5).

$$S(A,B) = HAR \times \frac{1 - GD}{SAF} \quad (5)$$

The HAR as shown in Eq. (6) is used to assess the degree of difference in height between two generalized Fuzzy numbers. This ratio decreases as difference between w_A and w_B increases and therefore reduces the similarity degree when there is a substantial difference between them. However, it may appear that consideration of HAR without square root may lead to the HAR being too insensitive to small differences between w_A and w_B. For example, assume w_A=0.4 and w_B=0.6 then the ratio will be 0.4/0.6=0.6667 which has great influence on the similarity degree even though the difference is not too high. But if we use the square root the ratio will be lower and equal to 0.8165, which in this case has a moderate influence on the similarity degree.

$$HAR = \sqrt{\min(w_A / w_B, w_B / w_A)} \quad (6)$$

The GD as shown in Eq. (7) is used to measure the geometric distance between two generalized Fuzzy numbers including the distance between their x-axis centroid. This distance increases as the difference between their elements increases.

$$GD = \frac{1}{5}\left(\sum_{i=1}^{4} |a_i - b_i| + |x_A - x_B| \right) \quad (7)$$

SAF is used to adjust the geometric distance, for example, if the two Fuzzy numbers have different shapes such as in case of one of the Fuzzy numbers is triangular and the other is trapezoidal or in case of unsymmetrical Fuzzy numbers, the similarity degree should decrease. This factor increases as the difference between y_A and y_B increases which means that two Fuzzy numbers have different shapes. In general the similarity degree decreases when SAF increases and HAR decreases which means that two generalized Fuzzy numbers have different shapes and different heights.

$$SAF = 1 + | y_A - y_B | \quad (8)$$

The (x_A, y_A) and (x_B, y_B) points in Eq. (9) and (10) represent the COG points of generalized Fuzzy numbers A and B respectively that are calculated as follows (Chen & Chen, 2008) :

$$y_A = \begin{cases} \dfrac{w_A \times \left(\dfrac{a_3 - a_2}{a_4 - a_1} + 2 \right)}{6}, \; if\ a_1 \neq a_4 \\ \dfrac{w_A}{2}, \;\; if\ a_1 = a_4 \end{cases} \quad (9)$$

$$x_A = \frac{y_A(a_3 + a_2) + (a_4 + a_1)(w_A - y_A)}{2w_A} \quad (10)$$

Below we consider some properties of such a measure in an attempt to motivate it further:

Property 1: Two generalized Fuzzy numbers $\tilde{A}$ and $\tilde{B}$ are identical *iff* $S(\tilde{A}, \tilde{B})=1$

Proof: Let $\tilde{A} = [a_1, a_2, a_3, a_4; w_A]$, $\tilde{B} = [b_1, b_2, b_3, b_4; w_B]$, $w_A, w_B \in [0,1]$,

(I) if $\tilde{A}$ and $\tilde{B}$ are identical this implies: $a_1 = b_1, a_2 = b_2, a_3 = b_3, a_4 = b_4, w_A = w_B, y_A = y_B$, then

$\sqrt{\min(w_A / w_B, w_B / w_A)} = 1$, $(1 + | y_A - y_B |) = 1$, and $\sum_{i=1}^{4} |a_i - b_i| = 0$,

Since $x_A = x_B$ then $| x_A - x_B | = 0$, then $S(\tilde{A}, \tilde{B}) = 1 * \dfrac{1 - \frac{1}{5}(0+0)}{1} = 1$

(II) Now let us prove that if $S(\tilde{A}, \tilde{B})=1$ then $\tilde{A}$ and $\tilde{B}$ are identical. This should be equivalent with: if $\tilde{A}$ and $\tilde{B}$ are not identical then $S(\tilde{A}, \tilde{B}) \neq 1$: Let us say $\tilde{A} \neq \tilde{B}$ then at least one of the elements in both Fuzzy numbers is different (i.e. $\exists_i\ a_i \neq b_i$ or $w_A \neq w_B$. Then $\sqrt{\min(w_B / w_A, w_A / w_B)} < 1$, or because $\exists_i\ a_i \neq b_i$ then $GD = \dfrac{1}{5}\left(\sum_{i=1}^{4} |a_i - b_i| + | x_A - x_B | \right) > 0$, and similarly $SAF = 1 + | y_A - y_B | \geq 1$. Thus, this implies that: $S(\tilde{A}, \tilde{B}) \neq 1$, and $0 \leq S(\tilde{A}, \tilde{B}) < 1$.

Property 2: $S(\tilde{A}, \tilde{B}) = S(\tilde{B}, \tilde{A})$

Proof: Since $\sqrt{\min(w_A / w_B, w_B / w_A)} = \sqrt{\min(w_B / w_A, w_A / w_B)}$,

$\sum_{i=1}^{4} |a_i - b_i| = \sum_{i=1}^{4} |b_i - a_i|$, $(1 + | y_A - y_B |) = (1 + | y_B - y_A |)$,

$| x_A - x_B | = | x_B - x_A |$, then $S(\tilde{A}, \tilde{B}) = S(\tilde{B}, \tilde{A})$

Property 3: if $\tilde{A}=[a,a,a,a;w_A]$, $\tilde{B}=[b,b,b,b;w_B]$ are two real value numbers then $S(\tilde{A},\tilde{B})$=1- | a-b | .

Proof: since $w_A = w_B$, then $\sqrt{\min(w_A/w_B, w_B/w_A)}$ =1. Since $y_A = y_B$ then $(1+|y_A - y_B|)$ =1, since $x_A = a$ and $x_B = b$ then $|x_A - x_B| = |a-b|$. Since $\sum_{i=1}^{4}|a_i - b_i| = (|a-b|+|a-b|+|a-b|+|a-b|) = (4\times|a-b|)$,

then $S(\tilde{A},\tilde{B}) = 1\times\frac{1-\frac{1}{5}(4\times|a-b|+|a-b|)}{1} = 1-|a-b|$

Example 1: suppose that A=[0.1, 0.3, 0.3, 0.5; 1] , B=[0.2, 0.4, 0.4, 0.7; 1] and C=[0.5, 0.7, 0.7, 0.9; 1] are three generalized triangular Fuzzy numbers as shown in Figure 3 . The COG point for each generalized Fuzzy number is calculated according to Eqs. 9 & 10, which resulted in COG(A)=(0.3,1/3), COG(B)=(0.4333,1/3), COG(C)=(0.7,1/3).

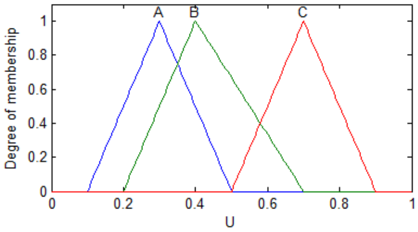

Figure 3. Three generalized Fuzzy numbers

The similarity degree between each two generalized Fuzzy numbers is computed as shown below. Observation from Figure 3 suggests that B is closer to A than C to A.

$$S(A,B) = \sqrt{\min(1/1,1/1)} \times \frac{1-\left(\frac{1}{5}\times(|0.1-0.2|+|0.3-0.4|+|0.3-0.4|+|0.5-0.7|)+|0.3\text{-}0.433|\right)}{1+|0.333-0.333|} = 0.8733$$

$$S(A,C) = \sqrt{\min(1/1,1/1)} \times \frac{1-\left(\frac{1}{5}\times(|0.1-0.5|+|0.3-0.7|+|0.3-0.7|+|0.5-0.9|)+|0.3\text{-}0.7|\right)}{1+|0.333-0.333|} = 0.6$$

$$S(B,C) = \sqrt{\min(1/1,1/1)} \times \frac{1-\left(\frac{1}{5}\times(|0.2-0.5|+|0.4-0.7|+|0.4-0.7|+|0.7-0.9|)+|0.433\text{-}0.7|\right)}{1+|0.333-0.333|} = 0.7267$$

5. Software prediction using Fuzzy numbers

The paper presents Fuzzy numbers as means to improve performance of analogy-based estimation at early stage of software development. In the introduction we mentioned different sources of uncertainty in software estimation such as: measurement error, model error, assumption error and scope error. In this paper we are concerned about uncertainty in the measurement and model because they are more likely related to the estimation by analogy at early stage. The first source of error is caused by measurement error during data collection as a result of human judgement which is often imprecise and vague. The second source of error is caused by the inability of the model to capture all details of the problem. For example, we may find that two projects are similar in terms of input domain (attributes) but their efforts are completely different. One principle reason appears to be that no effort estimation model includes all the factors that factually effort requires to accomplish the software project (Idri et al., 2001).

The proposed GFNSE model uses analogy together with the concept of generalized Fuzzy numbers. Using Fuzzy numbers allows us to explicitly model uncertainty in attribute values (in an effort to tolerate uncertainty in the final estimate). The proposed similarity measure and adaptation technique (explained later in section 5.4) are employed to develop GFNSE model. The procedure of prediction is explicitly explained in the following subsections. We should note here that "target project" is used to refer to the project under estimation and "comparative project" is used to refer to a project in the historical dataset.

5.1 Fuzzy Numbers Construction

The use of Fuzzy numbers in software EA requires determination of their spreads (degree of fuzziness). Fuzzy numbers can be constructed from either expert opinion or from data (Jowers et al., 2007). The former is totally subjective and depend on identifying pessimistic, optimistic and most likely values for each Fuzzy number, where the latter constructs Fuzzy numbers based on the structure of data (as we do in this paper).). One principle reason behind using the second approach is to model uncertainty explicitly during similarity measurement (Jorgensen, 2004).

Each real number at each attribute should be replaced by its corresponding Fuzzy number. In this regard the systematic triangular generalized Fuzzy numbers is our special interest in our study. Although the symmetric Fuzzy number does not take into account all kinds of estimation uncertainty (Yen et al., 1997), we still prefer to use it because it is rather simpler to find elements for when compared with the non-symmetric case. In this paper we propose an approach to derive the spread of Fuzzy number based on Fuzzy C-means (FCM) (Bezdek, 1981).

Because the information given in the dataset is insufficient to empirically determine spreads of Fuzzy numbers (i.e. the uncertainty interval is not defined), we suggest the use of the Fuzzy modelling based on FCM clustering algorithm to assess the degree of fuzziness in each real value. Like clustering algorithms which assign a data point to distinct cluster, FCM algorithm assigns membership values to each observation in all obtained clusters. Using Fuzzy modelling based on FCM, each attribute of a software project can be specified by the distribution of its possible values and represented in the form of Fuzzy clusters (also known as Fuzzy sets) (Bezdek, 1981). To build Fuzzy modelling based on FCM we used *genfis3* function that is implemented in MATLAB 2007a. *genfis3* is a MATLAB function used to construct a Fuzzy model based on the concept of Fuzzy clustering. It mainly uses FCM to derive membership values of all observations in each cluster. The generated membership values and clusters' centres are then used to construct Fuzzy model. The process of Fuzzy model construction can be understood by the following simple illustration. Suppose there are *N* data samples are described by 3 dimensional attributes (FA. FB and FC) as shown in Figure 4 which are clustered using FCM algorithm into 3 Fuzzy clusters as shown in Figure 5. These Fuzzy clusters are then used to construct their corresponding Fuzzy sets on each universe of discourse as shown in Figures 6, 7 and 8.

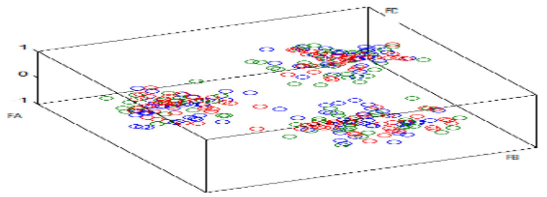

Figure 4. N data sample distributed in 3 dimensions.

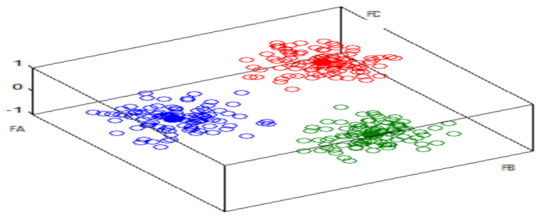

Figure 5. Clusters of N data sample.

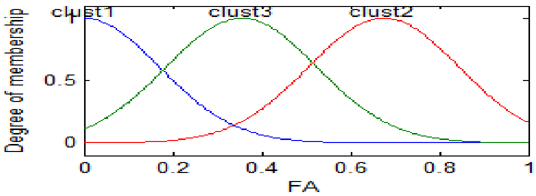

Figure 6. Membership functions for attribute FA.

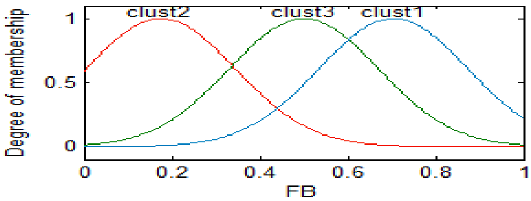

Figure 7 Membership functions for attribute FB.

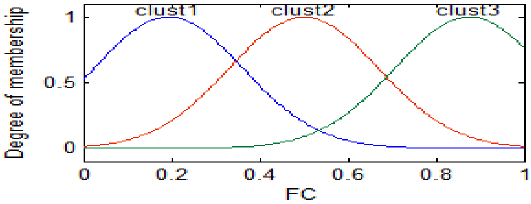

Figure 8 Membership functions for attribute FC.

The use of GFNSE requires prior determination of appropriate number of clusters that helps in deriving spreads of Fuzzy numbers. However, the appropriate number of Fuzzy clusters is influenced by dataset structure, number of observations, number and range of attribute values. To obtain suitable number of Fuzzy clusters we adopted the metric proposed by Xie and Beni (1991) to measure the compactness and coherence of Fuzzy clusters as shown in Eq. (11).

$$XB = \frac{\sum_{i=1}^{C} \sum_{k=1}^{N} (u_{ij})^2 \| C_i - x_k \|}{N * \min_{i,k} \| C_i - C_k \|} \qquad (11)$$

where C_i is the i^{th} center vector, u is the partition matrix, x_k is the k^{th} observation, and $\| \ \|$ is Euclidian distance. A small value of XB means a more compact and separate clustering. The goal should therefore to minimize XB in order to have more coherence in Fuzzy clusters. To find the correct number of Fuzzy clusters for each dataset we empirically assess the degree of compactness (XB) by varying number of clusters C from 2 to 10. The C that gives minimum XB is then chosen.

We assume that the Fuzzy sets obtained by FCM are a general form of Fuzzy number if it respects convex and normal properties, therefore we suggest making scaling between Fuzzy sets and real numbers in each attribute dimension. Figure 9 shows the process of spread determination of a Fuzzy number. We compute spread σ_{ik} for each i^{th} value at the k^{th} attribute as given in Eq. (12). The rationale behind this equation is the attempt to avoid small or too flat spread of Fuzzy number because some of Fuzzy clusters are too small or too flat.

$$\sigma_{ik} = \frac{1}{C}\sum_{j=1}^{C}\mu_j(p_i(k)) \times \left[\frac{x_i \times \delta_{jk}}{Center_{jk}}\right] \qquad (12)$$

where $\mu_j(p_i(k))$ is the membership value of i^{th} project at the k^{th} attribute in the j^{th} Fuzzy cluster. δ_{jk} and $Center_{jk}$ are the spread and center of j^{th} Fuzzy cluster at the k^{th} attribute. After determining the spread of a Fuzzy number the corresponding elements are given in Eq. (13) where b and c equal to the actual value, a=b- $\frac{\sigma_{ik}}{2}$, and d= b+ $\frac{\sigma_{ik}}{2}$

$$\tilde{P_i}(k) = \left[P_i(k) - \frac{\sigma_{ik}}{2},\ P_i(k),\ P_i(k),\ P_i(k) + \frac{\sigma_{ik}}{2};\ 1\right] \qquad (13)$$

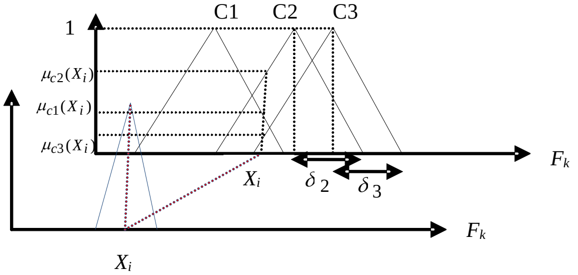

Figure 9. Fuzzy number coefficient determination

Example 2: suppose there are three Fuzzy clusters C1, C2 and C3 represented at the k^{th} attribute as shown in Figure 5. The centre and spread of each Fuzzy cluster is given as follows:

C1: (Centre1=0.2, $\delta_1 = 0.1$), C2: (Centre2=0.4, $\delta_2 = 0.15$), C3: (Centre3=0.5, $\delta_3 = 0.1$),

Given that the membership values of numeric data (X_i=0.32): $\mu_{C1}(X_i) = 0.3$, $\mu_{C2}(X_i) = 0.7$ and $\mu_{C3}(X_i) = 0.1$. The spread of corresponding Fuzzy number is calculated according to Eq. (13):

$$\sigma_{Xi} = \frac{1}{3}\left(0.3 \times \left[\frac{0.32 \times 0.1}{0.2}\right] + 0.7 \times \left[\frac{0.32 \times 0.15}{0.4}\right] + 0.1 \times \left[\frac{0.32 \times 0.1}{0.5}\right]\right) = 0.0461$$

As consequence, the parameters of the Fuzzy number X_i should be represented in the following sequence: [0.2969, 0.32, 0.32, 0.3431] according to Eq. (13).

5.2 Finding similarity between target project and each comparative project at k^{th} attribute.

In this section we employed the proposed similarity measure between Fuzzy numbers into software project similarity measurement. The similarity between the target project P_o and the comparative project P_i at the k^{th} attribute is defined as follows:

$$S(\tilde{p_o}(k), \tilde{p_i}(k)) = \sqrt{\min(w_{p_o(k)} / w_{p_i(k)}, w_{p_i(k)} / w_{p_o(k)})} \times \frac{1 - \frac{1}{5}\left(\sum_{j=1}^{4}\left|\tilde{p_{oj}}(k) - \tilde{p_{ij}}(k)\right| + |x_{p_o(k)} - x_{p_i(k)}|\right)}{1 + |y_{p_o(k)} - y_{p_i(k)}|} \qquad (14)$$

where:

- $\tilde{p}_o(k)$ and $\tilde{p}_i(k)$ are Fuzzy numbers.
- $\tilde{p}_{oj}(k)$: is the j^{th} parameter of the k^{th} attribute (Fuzzy number) of target project.
- $\tilde{p}_{\mathrm{ij}}(k))$: is the j^{th} parameter of the k^{th} attribute (Fuzzy number) of i^{th} comparative project.

The aggregated similarity between target project and comparative project on M attributes is computed as shown in Eq. (15).

$$S(\tilde{p}_o, \tilde{p}_i) = \frac{1}{M}\sum_{k=1}^{M} S(\tilde{p}_o(k), \tilde{p}_i(k)) \,, \quad (15)$$

5.3 Ranking closest projects

After calculating the aggregated similarity measure between target project and each individual comparative project, the similarity results are ranked to retrieve the closest project. The project with highest similarity has the greatest opportunity to contribute in the final estimate.

5.4 Deriving a new estimate for target project.

In most cases, using only the closest project to derive new estimate is not sufficient (Idri et al., 2001; Mendes et al, 2003a). It may lead to bad estimation accuracy because it is not likely that two similar projects that agreed on all attribute values should have the same effort value. Therefore, the estimation should be made by involving the projects that have high similarity degrees with the target project in order to contribute to better estimation accuracy. The proposed adaptation technique is based on generalized Fuzzy number operations. Assume the similarity degree between two projects is expressed as generalized Fuzzy number as follows:

$$S(\tilde{p}_o, \tilde{p}_i) = \left[S(\tilde{p}_o, \tilde{p}_i),\ \ S(\tilde{p}_o, \tilde{p}_i),\ \ S(\tilde{p}_o, \tilde{p}_i),\ \ S(\tilde{p}_o, \tilde{p}_i); 1 \right] \quad (16)$$

The normalized closest effort value E_i is converted into a generalized Fuzzy number $\tilde{E}_i$ as shown in Eq. (17):

$$\tilde{E}_i = \left[E_i - \frac{\sigma_i}{2},\ E_i,\ E_i,\ E_i + \frac{\sigma_i}{2}\ ;\ 1 \right] \quad (17)$$

The generalized Fuzzy number of predicted effort $\tilde{E}_o$ is calculated based on Fuzzy similarity adjustment as shown in Eq. (18). This equation utilizes Fuzzy numbers arithmetic operations to adjust retrieved effort values based on their similarity degrees.

$$\tilde{E}_o = \overset{K}{\underset{i=1}{\oplus}} \left[S(\tilde{p}_o, \tilde{p}_i) \otimes \tilde{E}_i \right]\ Ø\ \overset{K}{\underset{i=1}{\oplus}} \left[S(\tilde{p}_o, \tilde{p}_i) \right] \quad (18)$$

where K is the number of involved analogies. $\oplus$, $\otimes$ and Ø are the addition, multiplication and division Fuzzy numbers arithmetic operations respectively.

The COG of $\tilde{E}_o$ is computed to determine (x_o, y_o), then real value of predicted effort value $\hat{E}_o$ is calculated according to Eq. (19) (de-normalization). This equation is used to convert normalized value to original value based on the current set of effort *E.*

$$\hat{E}_o = x_o \times (\max(E) - \min(E)) + \min(E) \qquad (19)$$

Example 3: suppose a dataset has effort values range from 1000 to 4000 (i.e. min and max effort values). Assume we want to predict a project *p* using only the closest 3 analogies, the Table 3 depicts the most similar projects effort and their associated similarity degree. The similarity degrees and effort values are replaced by their corresponding Fuzzy numbers as explained previously. The calculation procedure that is used to estimate the effort of the target project is given below:

Table 1 illustrative example of adaptation technique

Effort	**Similarity degree**	**Effort as Fuzzy number**	**Similarity degree as Fuzzy number**
3100	**0.9**	[0.62, 0.65, 0.65, 0.7; 1]	[0.9, 0.9, 0.9, 0.9; 1]
2620	**0.85**	[0.43, 0.46, 0.46, 0.5; 1]	[0.85, 0.85, 0.85, 0.85; 1]
3440	**0.8**	[0.71, 0.73, 0.73, 0.75; 1]	[0.8, 0.8, 0.8, 0.8; 1]

$$\begin{aligned}\tilde{E}_o = \Big(\, & [0.9\times 0.62, 0.9\times 0.65, 0.9\times 0.65, 0.9\times 0.7; \min(1,1)] \oplus \\ & [0.85\times 0.43, 0.85\times 0.46, 0.85\times 0.46, 0.85\times 0.5\,; \min(1,1)] \oplus \\ & [0.8\times 0.71, 0.8\times 0.73, 0.8\times 0.73, 0.8\times 0.75\,; \min(1,1)]\ \Big) Ø \\ & [0.9+0.85+0.8,\ 0.9+0.85+0.8,\ 0.9+0.85+0.8,\ 0.9+0.85+0.8;\ \min(1,1)]\end{aligned}$$

$$\tilde{E}_o = \frac{[0.558, 0.585, 0.585, 0.63] \oplus [0.3655, 0.391, 0.391, 0.425] \oplus [0.568, 0.584, 0.584, 0.6]}{[2.55,\ 2.55,\ 2.55, 2.55]}$$

$$\tilde{E}_o = \frac{[1.4915, 0.156, 1.56, 1.655\,;\,1]}{[2.55,\ 2.55,\ 2.55, 2.55\,;\,1]} = [0.5849, 0.6118, 0.6118, 0.649;1]$$

The COG point of the obtained Fuzzy number $(x_o, y_o) = (0.6152, 0.333)$

The final estimate $\hat{E}_o = 0.6152 \times (4000 - 1000) + 1000 = 2845.6\ \ man-months$

6. Evaluation criteria

To assess the accuracy of the proposed estimation model, we have used the common evaluation criteria in the field of software cost estimation.

(I) Magnitude Relative Error (*MRE*) computes the absolute percentage of error between actual and predicted effort for each reference project.

$$MRE_i = \frac{|\,actual_i \text{ - } estimated_i\,|}{actual_i} \qquad (20)$$

(II) Mean magnitude relative error (*MMRE*) calculates the average of *MRE* over all reference projects. Despite of the widely used of *MMRE* in estimation accuracy, there has been a substantive discussion about efficacy of *MMRE* in estimation process. *MMRE* has been criticised that is unbalanced in many validation circumstances and leads often to overestimation (Shepperd and Schofield, 1997). Moreover, *MMRE* is not always reliable to compare between prediction methods because it is quite related to the measure of *MRE* spread (Foss et al., 2003). Therefore we used non-parametric statistical significance test to compare between the median of two samples based on absolute residuals, setting the confidence limit at 0.05. Since all absolute residuals were not normally distributed as confirmed by one-sample D'Agostino-Pearson test for non-normality, we used Mann Whitney U test of absolute residuals to investigate the statistical significance between different prediction models and Wilcoxon signed rank test to compare between paired absolute residuals.

$$MMRE = \frac{1}{N}\sum_{i=1}^{N} MRE_i \qquad (21)$$

(III) Since the *MMRE* is sensitive to an individual outlying prediction, when we have a large number of observations, we adopt median of *MREs* for the *n* projects (*MdMRE*) which is less sensitive to the extreme values of *MRE*.

$$MdMRE = \underset{i}{median}(MRE_i) \qquad (22)$$

(IV) *PRED* (ℓ) is used as complementary criterion to count the percentage of estimates that fall within less than or equal to ℓ of the actual values. The common used value for ℓ is 25%.

$$PRED(\ell) = \frac{\lambda}{N} \times 100 \qquad (23)$$

Where λ is the number of projects where $MRE_i \le \ell\%$, and N is the number of all observations. A software estimation model with lower *MMRE*, *MdMRE*, and higher *PRED(25%)* shows its derived estimates are more accurate than other models.

We also used Boxplot of absolute residuals to compare between different prediction techniques. The Boxplot shows the median as the central tendency of distribution, inter-quartile range and the outliers of individual models. The length of Boxplot from lower tail to upper tail shows the spread of the distribution. The length of box represents the range that contains 50% of observations. The position of median inside the box and length of Boxplot indicates the skewness of distribution. A Boxplot with a small box and long tails represents a very peaked distribution while a Boxplot with long box represents a flatter distribution.

7. Experimental Results

7.1 Design of Experiments

The proposed software effort estimation model GFNSE has been empirically examined through a series of evaluation studies using ISBSG, Desharnais, COCOMO, Kemerer, and Albrecht data sets. It is important to note that the proposed similarity measure is applicable only for normalized numerical data, because categorical attributes are not clearly suitable to be represented by Fuzzy numbers based on FCM. Therefore, the numerical attributes should be normalized to the same range in order to facilitate their comparison. In this regard, all numerical attributes were normalized to [0, 1] in which the degree of influence of i^{th} value at the j^{th} attribute is calculated according to Eq. (25).

Moreover, some of attributes in each data set that are not readily available at early stage (e.g. *KLOC*) were also left out.

$$x_i(j) = \frac{x_i(j) - \min(X_j)}{\max(X_j) - \min(X_j)} \tag{24}$$

where $j \in \{1,2,\ldots,M\}, i \in \{1,2,\ldots,n\}$

The empirical analyses have been conducted by using Jack knife procedure. Jack knifing procedure involves dividing the data set into multiple training and validation sets and aggregating the evaluation results across all validation sets. The evaluation procedure is repeated *n* times according to the number of observations. For each iteration, one project is held out once as test data and used exclusively to evaluate the performance of the data set that is trained on the remaining projects.

We also used non-parametric statistical significance test to compare between the median of two samples based on absolute residuals, setting the confidence limit at 0.05. Since all absolute residuals were not normally distributed as confirmed by one-sample D'Agostino-Pearson test for non-normality, we used Mann Whitney U test to investigate the statistical significance between different prediction models and Wilcoxon signed rank test to compare between paired absolute residuals.

To show the performance of GFNSE against other prediction methods, we conducted a number of experiments with reference to prediction accuracy in order to compare between GFNSE and two other popular existing prediction methods: Case Based Reasoning (CBR) and Stepwise Regression (SR). The choice of such prediction methods is based on the different strategies they use to make estimate. To assess the accuracy of the predictions generated by SR and CBR, a Jackknife validation strategy was used as for GFNSE. The attributes that are considered not readily available at early stage are left out for both CBR and SR.

For CBR we used ANGEL tool (Shepperd & Schofield, 1997) with the following configurations: (1) using exhaustive attribute subset election, (2) setting Euclidean distance as similarity measure, and (3) no adaptation technique is being used. This has the advantage of reducing user interactions in terms of configuring estimation by analogy method.

For SR, it is important to make sure that assumptions related to using stepwise regression are not violated before building effort prediction model (Mendes et al., 2003c). For example, skewed numerical variables need to be transformed such that they resemble more closely a normal distribution. The one-sample D'Agostino-Pearson test (*D-P test*) was used to check if all size variables such as (adjusted functions points, raw function points, transactions, entities and KLOC) are normally distributed. In case if anyone is not, so it was transformed to a natural logarithmic scale to approximate a normal distribution (Mendes et al., 2003c). Once transformed, its distributions are re-checked again to confirm that it is normally distributed. The logarithmic transformation ensures that the resulting model goes through the origin on the raw data scale. It also caters for both linear and non-linear relationships between size and effort. Further, all categorical attribute were converted into appropriate dummy variables. Above all, all necessary pre-requested tests such as normality tests are performed once before running empirical validation which resulted in a general regression model. Then, in each jack-knife iteration a different regression model, that resembles general regression model in the structure, is built based on the training data set and then the prediction of test project is made on training data set.

7.2 ISBSG data set

The analysis presented in this section is based on ISBSG repository (release 10 January 2007), which currently contains more than 4000 software projects gathered from different worldwide software development companies. All projects involved in the ISBSG repository are described by several numerical and categorical attributes. In order to assess the efficiency of the proposed similarity measures on software cost estimation we have selected a subset of 9 useful early numerical attributes including ‘*AFP*’, ‘*input_count*’, ’*output_count*’, ’*enquiry_count*’, ’*file_count*’, ’*interface_count*’, ’*add_count*’ , ‘*delete_count*’ and ’*changed_count*’. Since many projects have missing values only 500 projects with quality rating “A” are considered.

The results presented in this section have been obtained by applying Jackknife validation strategy. Particularly, the analysis part of this investigation show that the appropriate number of Fuzzy clusters is varying in each Jackknife iteration according to training data set structure, but approximately 6 clusters is the most influential value among all iterations. The performance figures of GFNSE in terms of *MMRE, MdMRE*, and *PRED (25)%* are presented in Table 2 with varying the number of respective analogies (i.e. *K*=1, 2, 3, 4, 5). Upon results analysis of the empirical validation, GFNSE(*K*=3) shows slightly better estimation accuracy than GFNSE(*K*=2) in terms of *MMRE* and *MdMRE*, but GFNSE(*K*=2) produces remarkable accuracy in terms of *PRED(25)%*. The GFNSE(*K*=1) produces slightly the worst results, suggesting that at least for ISBSG data set, using one analogy is unlikely to be the most adequate choice. Among the various numbers of analogies, we noticed in general that using 2 and 3 analogies based on our proposed adaptation technique would be the best choice for ISBSG and produced better results than using only closest analogy. In general, all results for GFNSE were relatively good, if we consider that *MMRE* ≤ 25% and *PRED(25)%* ≥ 75% suggest good accuracy level. Further, the difference between best and worst accuracy is remarkable and indication of the performance of our proposed adaptation technique. The values in bold represent the best accuracy obtained when using 3 analogies which will be used later to compare GFNSE to CBR and SR methods as shown in Table 3.

Table 2 Performance figures for ISBSG with different number of analogies

Model	*MMRE*	*MdMRE*	*PRED(25)%*
GFNSE (*K*=1)	37.86	25.99	54.06
GFNSE (*K*=2)	29.54	18.29	61.98
GFNSE (*K*=3)	**28.55**	**17.80**	**59.80**
GFNSE (*K*=4)	33.88	21.78	54.65
GFNSE (*K*=5)	32.82	21.46	55.64

To confirm the obtained performance figures and show whether the predictions generated by any GFNSE model (i.e. according to the number of analogies used) are significantly different from another, we used Wilcoxon signed rank test of paired absolute residuals. The results in Table 3 indicate that even though GFNSE (*K*=2) and GFNSE (*K*=3) obtained relatively similar prediction accuracy, surprisingly there is a statistical significance between them. We also noticed a statistical significance between GFNSE(*K*=3) and other GFNSE models, suggesting that, there is difference if the predications generated using GFNSE(*K*=3) than those compared models. On the other hand, we did not find any statistical significance between GFNSE(*K*=1) and GFNSE(*K*=2), or between GFNSE(*K*=1) and GFNSE(*K*=5), even though their relative prediction accuracy is remarkable.

Table 3 Wilcoxon signed rank test for paired absolute residuals over ISBSG

Model	Wilcoxon test	Model	Wilcoxon test
GFNSE(*K*=1) vs. GFNSE(*K*=2)	1.58	GFNSE(*K*=2) vs. GFNSE(*K*=4)	-1.60
GFNSE(*K*=1) vs. GFNSE(*K*=3)	3.72**	GFNSE(*K*=2) vs. GFNSE(*K*=5)	-0.92
GFNSE(*K*=1) vs. GFNSE(*K*=4)	0.064	GFNSE(*K*=3) vs. GFNSE(*K*=4)	-3.7**
GFNSE(*K*=1) vs. GFNSE(*K*=5)	0.534	GFNSE(*K*=3) vs. GFNSE(*K*=5)	-3.2**
GFNSE(*K*=2) vs. GFNSE(*K*=3)	2.44*	GFNSE(*K*=4) vs. GFNSE(*K*=5)	0.6225
Note: *statistically significant at 95%, ** statistically significant at 99%			

Table 4 shows results of comparing GFNSE to CBR and SR using best variants from Table 6.3, following up the similar validation strategy using Jackknife validation. Before building stepwise regression model, the *D-P test* found that most of the size attributes should be transformed to natural logarithmic scale. The resulted general stepwise regression model rejects the majority of the attributes as not contributing significantly to a model, only two significant attributes namely, '*AFP'* and '*ADD'* were involved in the model as shown in Eq. (25). The adjusted R^2 of 0.21 suggests that the model was not good with only 21% of the variation in effort being explained by variation in '*AFP* and '*ADD*'. The next step is to use the identified attributes in order to build a similar stepwise regression models for each training data set throughout Jackknife iterations, and accordingly predict each test project.

$$Ln(Effort) = 5.9318 + 0.261 \times Ln(AFP) + 0.066 \times Ln(ADD) \qquad (25)$$

The results of comparisons revealed that the GFNSE produced superior prediction accuracy in terms of all performance figures. This comes as no surprise as CBR uses Euclidean distance to assess the similarity degree between two project values, which as far as we know it is influenced by uncertainty (Idri et al., 2001) and extreme outliers. The similarity degree is amplified when a project with extreme values is assessed against observed project. Later this project will be excluded from similarity order in spite of its effort is more predictor. Moreover, each attribute in similarity measurement has the same degree of impact. Therefore the more correlated attribute will have the same influence as less correlated attribute. If we look at the worst accuracy obtained by GFNSE, it still outperforms SR and CBR with *MMRE* of 37.86%. Therefore we can conclude that the most accurate effort predictions are obtained from GFNSE model. These results are also confirmed by Mann Whitney U test as depicted in Table 5. The test shows, unsurprisingly, predictions based on GFNSE model presented statistically significant accurate estimations, measured using absolute residuals, suggesting that, based on the ISBSG data set characteristics it would make difference if predictions were generated using GFNSE or other models.

Table 4 Performance figures of comparing GFNSE to CBR and SR over ISBSG

Model	*MMRE%*	*MdMRE%*	*PRED(25)%*
GFNSE	**28.55**	**17.80**	**59.80**
CBR	52.32	30.23	42.71
SR	48.75	38.29	36.80

Table 5 Comparison of techniques over ISBSG, using Mann Whitney U test

Models	Mann Whitney U test
GFNSE vs. CBR	-4.8**
GFNSE vs. SR	-10.22**
CBR vs. SR	-3.09*

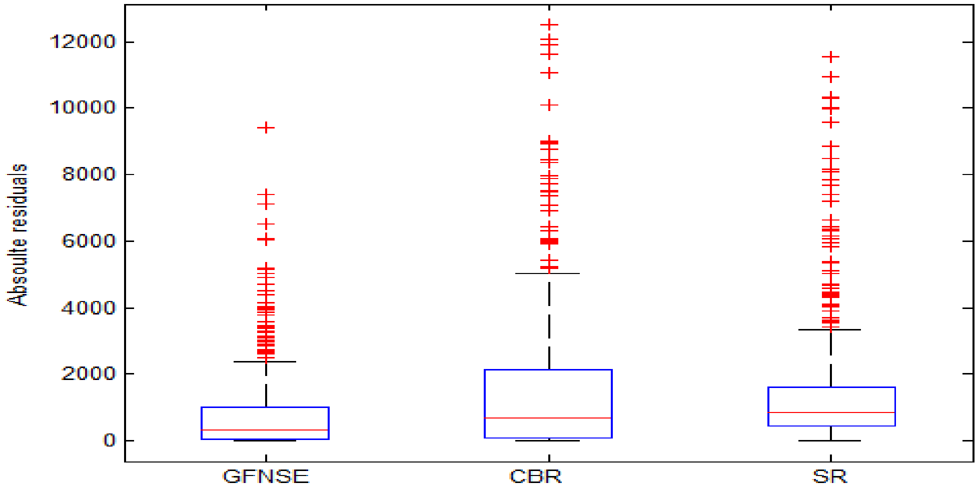

Figure 10 Boxplots of absolute residuals of GFNSE, CBR and SR over ISBSG

The results of statistical significance test are also confirmed by the results of Boxplot of absolute residuals as suggested by Kitchenham et al. (2001). The Boxplot of absolute residuals as shown in Figure 10 may provide a better insight on the effectiveness of a prediction model. Figure 10 shows that CBR produced the worst estimates with many extreme values of absolute residual. This problem may be caused by using irrelevant information (not representative training data), unrelated attributes and noisy data. However, the box of GFNSE overlays the lower tail which shows that the absolute residuals are skewed towards the minimum value and also presents accurate estimation than other three models. The range of absolute residuals of GFNSE is much smaller than absolute residuals of CBR which also presents smaller variance. The median of GFNSE is smaller than median of other models which revealed that at least half of the predictions of GFNSE are more accurate than those generated by other models.

7.3 COCOMO data set

Data set COCOMO was frequently used for validating various effort estimation methods. It includes 60 software projects that are described by 17 cost drivers (independent attributes) in conjunction with an actual effort. The actual effort in the COCOMO dataset is measured by person-month which represents the number of months that one person need to develop a given project. Despite the fact that the COCOMO datasets are now over 25 years old, it is still commonly used to assess the comparative accuracy of new techniques.

Applying GFNSE on COCOMO with using the same validation strategy as for ISBSG indicates that approximately 9 Fuzzy clusters are the influential value throughout Jackknife iterations. Note that *'LOC'* attribute has been left out, because it is not generally available at early stage. The performance figures in Table 6 show that GFNSE produced good estimates with *MMRE* touches minimum acceptable accuracy (i.e. *MMRE* of 25%). Like ISBSG data set, the worst accuracy has been obtained when using one and four analogies. This does not assume that the adaptation technique is not efficient but it confirms that the choice of number of analogies is a matter. However, regarding the effect of number of analogies on the prediction accuracy we can generally observe that the adaptation technique works well when two or three analogies are used. The difference between best and worst accuracy is also remarkable and confirmed by statistical significance tests that are shown in Table 7. This illustrates that the adaptation technique based on Fuzzy numbers performs better than classical adaption used in CBR such as mean and median or linear size adjustment (Mendes et al., 2003a; 2003b). The values in bold represent the best obtained accuracy for which the comparisons between GFNSE with CBR and SR are performed.

Table 6 Performance figures for COCOMO with different analogy numbers

K	*MMRE*	*MdMRE*	*PRED(25)%*
GFNSE (*K*=1)	60.39	50.54	26.67
GFNSE (*K*=2)	48.30	22.47	51.67
GFNSE (*K*=3)	***33.37***	***20.36***	***62.33***
GFNSE (*K*=4)	56.97	44.34	30.00
GFNSE (*K*=5)	53.89	31.00	31.67

The significance test based on Wilcoxon signed rank test revealed that difference between GFNSE(*K*=2) and GFNSE(*K*=3) is not statically significance. Suggesting that it would make no difference if predictions were generated using GFNSE(*K*=2) or GFNSE(*K*=3), Even though their prediction accuracy are very close. We also noticed statistical significance between GFNSE(*K*=3) and (GFNSE(*K*=1), GFNSE(*K*=4) and GFNSE(*K*=5)), which confirm that predictions generated by GFNSE(*K*=3) are significantly different from those generated by GFNSE(*K*=1) or GFNSE(*K*=4) or GFNSE(*K*=5), and relatively similar for those generated by GFNSE(*K*=2).

Table 7 Wilcoxon signed rank test for paired absolute residuals over COCOMO

Model	Wilcoxon test	Model	Wilcoxon test
GFNSE(*K*=1) vs. GFNSE(*K*=2)	1.81	GFNSE(*K*=2) vs. GFNSE(*K*=4)	-1.45
GFNSE(*K*=1) vs. GFNSE(*K*=3)	2.61**	GFNSE(*K*=2) vs. GFNSE(*K*=5)	-1.12
GFNSE(*K*=1) vs. GFNSE(*K*=4)	0.428	GFNSE(*K*=3) vs. GFNSE(*K*=4)	-1.91**
GFNSE(*K*=1) vs. GFNSE(*K*=5)	0.7243	GFNSE(*K*=3) vs. GFNSE(*K*=5)	-1.83*
GFNSE(*K*=2) vs. GFNSE(*K*=3)	0.126	GFNSE(*K*=4) vs. GFNSE(*K*=5)	0.40
Note: *statistically significant at 95%, ** statistically significant at 99%			

The general model of stepwise regression model is represented in Eq. (26) which again rejected most of the attributes as not significantly contributing to the regression model. This model was built after running appropriate test to check whether the transformation to natural logarithmic scale is necessary. Only *Effort* attribute required transformation. The adjusted R^2 of 0.18 suggests that the general stepwise regression model is very poor. However, the obtained results of comparison show that GFNSE produced relatively better estimation accuracy than CBR and SR in respects of *MMRE, MdMRE,* but not in terms of *PRED(25)%*. It also seems that CBR yielded better results than SR in terms *MMRE* in spite of the COCOMO model was originally built on regression model. The results are also confirmed by statistical significance test using Mann Whitney U test which indicates that the absolute residuals of GFNSE are statistically significant and different from those of SR, and not significantly different from CBR as shown in Table 9.

$$Ln(Effort) = 2.93 - 3.813 \times PCAP + 5.94 \times TURN \tag{26}$$

Table 8 Performance figures of comparing GFNSE to CBR and SR over COCOMO

Model	*MMRE%*	*MdMRE%*	*PRED(25)%*
GFNSE	***33.37***	***20.36***	***62.33***
CBR	47.3	33.8	35.0
SR	96.6	82.4	23.1

Table 9 Comparison of techniques over COCOMO, using Mann Whitney U test

Models	Mann Whitney U test
GFNSE vs. CBR	-0.5
GFNSE vs. SR	-4.067**
CBR vs. SR	-1.5*

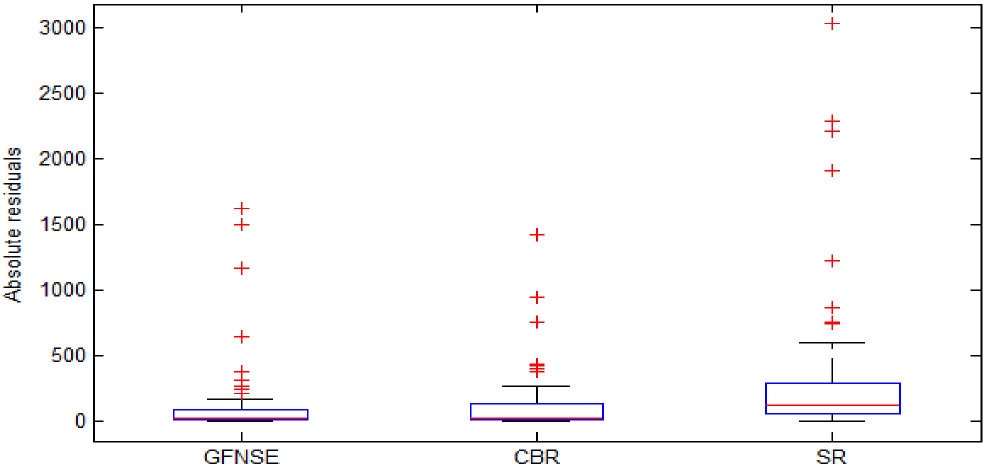

Figure 11 Boxplots of absolute residuals of GFNSE , CBR and SR over COCOMO

From Figure 11 we can come to conclude that GFNSE generated relatively accurate predictions than others, and corroborated by the following figures: (1) the box length of GFNSE is smaller than others which demonstrates reduced variability in absolute residuals. (2) The box of GFNSE and CBR overlay the lower tails which also present accurate estimation than SR model because the absolute residuals are skewed towards the minimum value. (3) The median of GFNSE and CBR are smaller than median of SR which confirms that at least half of the predictions of GFNSE and CBR are more accurate than those generated by other models. Lastly, according to the Boxplot, the SR gave the worst absolute residual value with many extreme outliers.

7.4 Desharnais data set

The Desharnais dataset consists of 81 software projects collected from Canadian software houses (Boetticher et al., 2008). This dataset is described by 11 attributes, for which one of them is the effort measured in '1000 person-hours', whereas the others are used to represent software projects: '*TeamExp*', '*ManagerExp*', '*YearEnd*', '*Duration*', '*Transactions*', '*Entities*', '*AdjFP*', '*AdjFactor*', '*RawFP*', and '*Dev.Env*'. Unfortunately, 4 projects out of 81 contain missing values therefore we excluded these projects which resulted in 77 complete software projects.

For Desharnais data set, the results obtained in Table 10 are relatively similar to those obtained for COCOMO and ISBSG data sets. Two attribute namely, '*Duration*' and '*Dev.Env*' have been left out for GFNSE as '*Duration*' is dependent attribute and '*Dev.Env*' is categorical attribute. Initially, the appropriate number of Fuzzy cluster was approximately 9 clusters, which is an indication to the need for sufficient number of clusters to clearly represent data. *MMRE* evaluation criterion suggests that GFNSE(*K*=2) tends to be more accurate than other GFNSE models. Similar to ISBSG and COCOMO data set, among all number of analogies involved in this experiment the worst estimate was obtained when applying only one analogy. As an aside, the difference between best accuracy (GFNSE(*K*=2)) and worst prediction accuracy (GFNSE(*K*=1)) is relatively significant. To conclude, the adaptation technique works particularly well over Desharnais data set especially for 2, 3, and 5 analogies, as it does for COCOMO and ISBSG.

Table 10 Performance figures for Desharnais with different analogy numbers

K	*MMRE*	*MdMRE*	*PRED(25)%*
GFNSE (*K*=1)	40.86	23.08	55.84
GFNSE (*K*=2)	***26.89***	***19.32***	***64.94***
GFNSE (*K*=3)	31.93	23.82	55.84
GFNSE (*K*=4)	35.91	24.77	50.65
GFNSE (*K*=5)	32.67	20.44	59.74

Table 11 Wilcoxon signed rank test for paired absolute residuals over Desharnais

Model	Wilcoxon test	Model	Wilcoxon test
GFNSE(*K*=1) vs. GFNSE(*K*=2)	2.87*	GFNSE(*K*=2) vs. GFNSE(*K*=4)	-1.32
GFNSE(*K*=1) vs. GFNSE(*K*=3)	1.056*	GFNSE(*K*=2) vs. GFNSE(*K*=5)	-0.051
GFNSE(*K*=1) vs. GFNSE(*K*=4)	0.01	GFNSE(*K*=3) vs. GFNSE(*K*=4)	0.09
GFNSE(*K*=1) vs. GFNSE(*K*=5)	0.83	GFNSE(*K*=3) vs. GFNSE(*K*=5)	0.63
GFNSE(*K*=2) vs. GFNSE(*K*=3)	-0.763	GFNSE(*K*=4) vs. GFNSE(*K*=5)	0.7625
Note: *statistically significant at 95%, ** statistically significant at 99%			

Before we constructed stepwise regression model for Desharnais data set, the categorical attribute (‘*Dev.Env*’) was converted into two dummy variables (*L1*and *L2*). The attribute *‘Duration’* has been left out for both CBR and SR as it is not readily available at early stage of software development. The resulted regression model as represented in Eq. 27 involved only four attributes for which two of them are the dummy variables. The dependent attribute (*Effort*) and independent attribute (*AdjFP*) were both transformed to natural logarithmic scale because they are not normally distributed, as confirmed by *D-P* normality test. The goodness of fit for regression model was 0.77 which suggests that the regression model is relatively good. However, Table 12 summarizes the aggregated results of applying GFNSE, SR and CBR on Desharnais data set. The results show that all prediction models produced good accuracy, but GFNSE is somewhat better especially in terms of *PRED(25)%*.

The performance of GFNSE is also confirmed by Mann Whitney U test. The significance test results indicate that there is statistical significance between predictions generated by GFNSE and other prediction models. But we did not find any statistical significance between CBR and SR which confirms that both models generate roughly equivalent predictions.

$$Ln(Effort) = 4.4 + 0.97 \times Ln(AdjFP) - 1.34 \times L1 - 1.37 \times L2 \quad (27)$$

Table 12 Performance figures of comparing GFNSE to CBR and SR over Desharnais

Model	*MMRE%*	*MdMRE%*	*PRED(25)%*
GFNSE	**26.89**	**19.32**	**64.94**
CBR	38.2%	30.8%	42.9%
SR	34.6%	28.6%	45.5%

Table 13 Comparison of techniques over Desharnais, using Mann Whitney U test

Models	Mann Whitney U test
GFNSE vs. CBR	-2.42*
GFNSE vs. SR	-2.15*
CBR vs. SR	-1.81

Figure 12 throws up some interesting results: (1) CBR and SR have relatively similar box length which demonstrates that at least half of predictions for CBR and SR at the same accurate level. (2) The larger inter-quartile of CBR and SR indicates a high dispersion of the absolute residuals. (3) The median of GFNSE is smaller than median of other models which shows that at least half of the predictions of GFNSE are more accurate than those generated by other models. (4) The box length of GFNSE is much smaller than others which demonstrate reduced variability in absolute residuals.

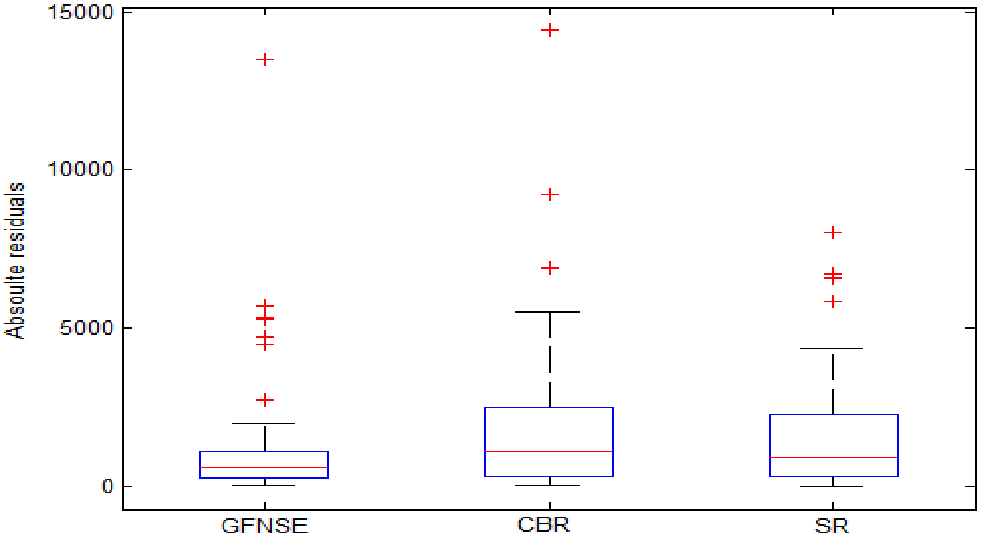

Figure 12 Boxplots of absolute residuals of GFNSE, CBR and SR over Desharnais

7.5 Albrecht data set

The Albrecht dataset (Albrecht, 1983) contains 24 software projects were developed by using third generation languages such as COBOL, PL1, etc. The dataset is described by one dependent attribute called 'work hours' which represents the corresponding effort in 1000 hour, and seven independent attributes: '*input count*', '*output count*', '*query count*', '*file count*', '*line of code*' ,'*RawFP*', and '*function points*'. 18 projects were written in COBOL, 4 projects were written in PL1 and the rest were written in data base management languages.

The analysis part of this examination show that in almost all jackknife iterations the corresponding training data sets require approximately 7 to 9 clusters to derive spreads of Fuzzy numbers, even though the size of data set is too small. This suggests that when number of clusters increases, the possibility of an observation to belong to more than one cluster is high, so the width of Fuzzy number becomes smaller because the width of Fuzzy clusters will be small and more compact as well.

However, the prediction accuracy results are not entirely superior and especially when using one and four analogies. One principle reason may be related to the existing of irrelevant or redundant attributes. This can be often solved by involving suitable attribute subset algorithm such as exhaustive search algorithm that is implemented in ANGEL tool (Shepperd & Schofield, 1997). We will leave this issue for further experiments in future work. Particularly, the predictions generated by GFNSE(*K*=2) and GFNSE(*K*=3) tend to be more accurate than other GFNSE models, suggesting that the use of adaptation technique with 2 or 3 analogies outperforms GFNSE without adaptation. Further, the difference between best and worst prediction accuracy is also remarkable. The best variant from Table 14 is selected to compare GFNSE with SR and CBR.

Table 14 Performance figures for Albrecht with different analogy numbers

K	*MMRE*	*MdMRE*	*PRED(25)%*
GFNSE (*K*=1)	81.40	33.17	37.50
GFNSE (*K*=2)	54.75	31.47	41.67
GFNSE (*K*=3)	*50.08*	*30.75*	*50.00*
GFNSE (*K*=4)	81.37	30.49	33.33
GFNSE (*K*=5)	76.88	30.01	41.67

The statistical significance test results between GFNSE models are presented in Table 15. The results obtained confirm that the predictions generated by GFNSE(K=3) are significantly different from those generated by GFNSE(K=1), GFNSE(K=4) and GFNSE(K=5). In contrast, we did not any statistical significance between GFNSE(K=2) and GFNSE(K=3) which means that there is no difference if the predictions generated by GFNSE(K=2) or GFNSE(K=3).

Table 15 Wilcoxon signed rank test for paired absolute residuals over Albrecht

Model	Wilcoxon test	Model	Wilcoxon test
GFNSE(*K*=1) vs. GFNSE(*K*=2)	1.17	GFNSE(*K*=2) vs. GFNSE(*K*=4)	-0.84
GFNSE(*K*=1) vs. GFNSE(*K*=3)	-3.16*	GFNSE(*K*=2) vs. GFNSE(*K*=5)	-1.02
GFNSE(*K*=1) vs. GFNSE(*K*=4)	0.4227	GFNSE(*K*=3) vs. GFNSE(*K*=4)	-3.81**
GFNSE(*K*=1) vs. GFNSE(*K*=5)	0.26	GFNSE(*K*=3) vs. GFNSE(*K*=5)	-2.1*
GFNSE(*K*=2) vs. GFNSE(*K*=3)	-0.44	GFNSE(*K*=4) vs. GFNSE(*K*=5)	-0.13
Note: *statistically significant at 95%, ** statistically significant at 99%			

Table 16 depicts the obtained results of comparison between GFNSE and other prediction techniques over Albrecht data set. The stepwise regression model involved only one significant independent attribute (*RawFP*) without need for natural logarithmic transformation. The resulted model as shown in Eq. (28) gives an excellent R^2 value of 0.902, but still leads to a poor level of accuracy in terms of *MMRE* (61.24%) because the model is not very good for small effort values. However, the results demonstrate that GFNSE yielded better estimates than others methods in terms of lower *MMRE* and *MdMRE*. Regarding *PRED(25)*, which measure the number of individual estimates that has *MRE* value less than 25%, GFNSE had the highest value among other models, these results are also confirmed by Boxplot of residuals. Regarding statistical significant using Mann Whitney U test, surprisingly, we did not find any statistical significance between GFNSE and other prediction models. This suggests that there is no significant difference if predictions were generated by GFNSE or SR and CBR.

$$Effort = -16.203 + 0.06 \times RawFP \tag{28}$$

Table 16 Performance figures of comparing GFNSE to CBR and SR over Albrecht

Model	*MMRE%*	*MdMRE%*	*PRED(25)%*
GFNSE	50.08	30.75	50.00
CBR	63.5	38.9	33.3
SR	61.24	32.3	37.5

Table 17 Comparison of techniques over Albrecht, using Mann Whitney U test

Models	Mann Whitney U test
GFNSE vs. CBR	-1.7
GFNSE vs. SR	-1.15
CBR vs. SR	0.34

Boxplots of absolute residuals (Figure 13) suggest that GFNSE gives the best prediction accuracy, confirmed by the following findings: (1) the box length of GFNSE is smaller than others which revealed reduced variability in absolute residuals. (2) The median of GFNSE is smaller than median of other models which shows at least half of the predictions of GFNSE are more accurate than those generated by other models. (3) The range of absolute residuals for SR and CBR are larger than absolute residuals for GFNSE model which means high dispersion.

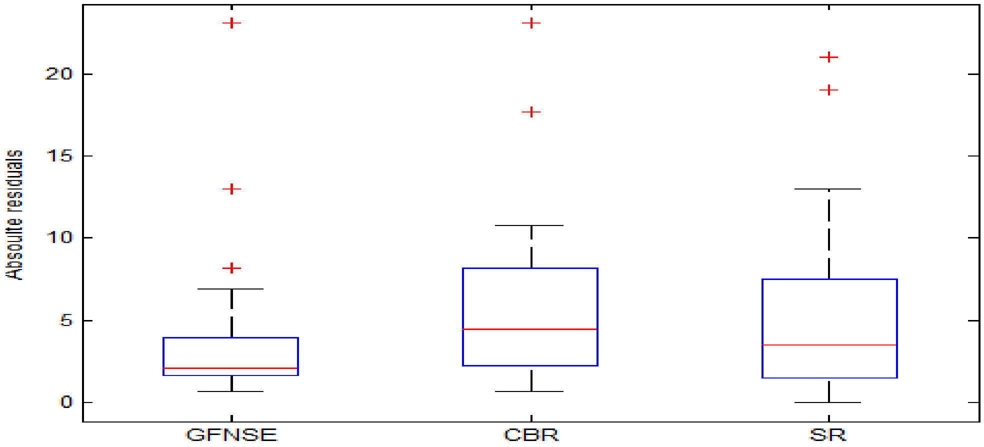

Figure 13 Boxplots of absolute residuals of GFNSE, CBR and SR over Albrecht

7.6 Kemerer data set

The Kemerer dataset includes 15 software projects described by 5 independent attributes and one dependent attribute. The independent attributes are represented by 2 categorical ('*software*', '*hardware*') and 3 numerical attributes: '*RawFp*', '*KSLOC* and '*AdjFP*. The effort attribute is measured by 'man-months'.

The attribute '*KSLOC*' has been left out as not readily available before prediction is required at beginning of the project. Table 18 summarizes results obtained by applying GFNSE with different number of analogies on Kemerer. As in other data sets, the best prediction accuracy obtained when applying adaptation technique. Particularly, the best obtained accuracy was produced when using 3 analogies. The difference between best and worst accuracy are remarkable. The results obtained for

Wilcoxon signed rank test confirm that the predictions generated by GFNSE(K=3) are not significantly different than those generated by GFNSE(K=1), GFNSE(K=4), and GFNSE(K=5).

Table 18 Performance figures for Kemerer with different analogy numbers

K	*MMRE*	*MdMRE*	*PRED(25)%*
GFNSE (*K*=1)	82.20	48.75	20.00
GFNSE (*K*=2)	64.53	46.35	33.33
GFNSE (*K*=3)	***55.65***	***24.24***	***53.33***
GFNSE (*K*=4)	79.17	61.4	26.67
GFNSE (*K*=5)	76.06	37.0	26.67

The best variants from Table 18 are selected as the candidates for comparisons of GFNSE to CBR and SR methods as shown in Table 20. The developed stepwise regression model as shown in Eq. (29) involved transformation of *Effort and AdjFP* attributes. This model presents moderate adjusted R^2 value of 0.667. Nevertheless, we can observe that the GFNSE generated accurate predictions than SR and CBR. These superior results are confirmed by Mann Whitney U test as shown in Table 21. Unsurprisingly, predictions based on GFNSE model presented statistically significant accurate estimations, measured using absolute residuals. Particularly, we found statistical significance between GFNSE and (CBR, and SR) over Kemerer, suggesting that, there is difference if the predications generated using GFNSE or other models. Further, we did not find a statistical significance between CBR and SR which indicates that there is no difference if predictions were generated by CBR or SR. These superior results are also confirmed by Boxplot of absolute residuals in Figure 14.

Table 19 Wilcoxon signed rank test for paired absolute residuals over Kemerer

Prediction models	Wilcoxon test	Model	Wilcoxon test
GFNSE(*K*=1) vs. GFNSE(*K*=2)	0.705	GFNSE(*K*=2) vs. GFNSE(*K*=4)	-0.660
GFNSE(*K*=1) vs. GFNSE(*K*=3)	0.788	GFNSE(*K*=2) vs. GFNSE(*K*=5)	-0.660
GFNSE(*K*=1) vs. GFNSE(*K*=4)	0.124	GFNSE(*K*=3) vs. GFNSE(*K*=4)	-0.456
GFNSE(*K*=1) vs. GFNSE(*K*=5)	0.540	GFNSE(*K*=3) vs. GFNSE(*K*=5)	0.166
GFNSE(*K*=2) vs. GFNSE(*K*=3)	0.00	GFNSE(*K*=4) vs. GFNSE(*K*=5)	0.040
Note: *statistically significant at 95%, ** statistically significant at 99%			

$$Ln(Effort) = -1.057 + 0.9 \times Ln(AdjFP) \tag{29}$$

Table 20 comparison of GFNSE to CBR and SR over Kemerer

Model	*MMRE%*	*MdMRE%*	*PRED(25)%*
GFNSE	***55.65***	***24.24***	***53.33***
CBR	63.8	33.33	40.00
SR	161.73	74.88	6.7

Table 21 Comparison of techniques over Kemerer, using Mann Whitney U test

Prediction models	**Mann Whitney U test**
GFNSE vs. CBR	-1.62*
GFNSE vs. SR	-2.16*
CBR vs. SR	-0.66

Boxplots of residuals (Figure 14) suggest that GFNSE gives the best prediction accuracy, confirmed by the following figures: (1) the box length of GFNSE is much smaller than others which demonstrates reduced variability in absolute residuals. (2) The median of GFNSE is smaller than median of other models which also indicates that at least half of the predictions of GFNSE are more accurate than those generated by other models.

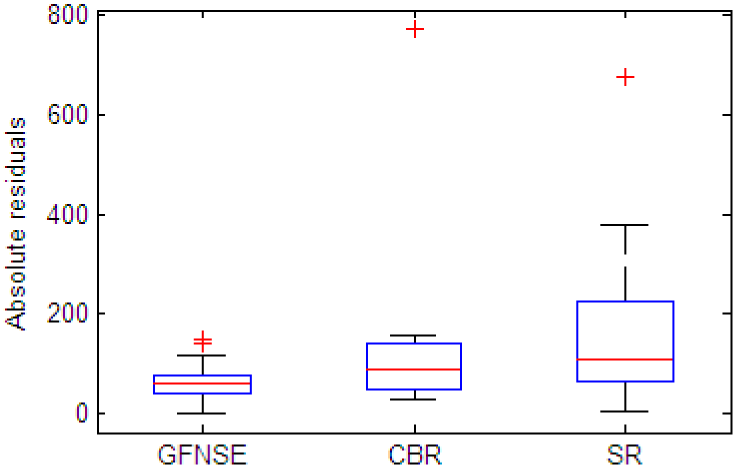

Figure 14 Boxplots of absolute residuals of GFNSE, CBR and SR over Kemerer

8. Discussion of results

Despite a great deal of research activity, predicting early stage effort for software development projects with any acceptable degree of accuracy remains challenging. One of our growing concerns is that of the uncertainty associated with available early data. This paper proposed a new software effort estimation technique based on Fuzzy Numbers that is entirely automatable and effective for early stage effort estimation. The proposed method can be now applied in the absence of attribute selection and where attributes are negatively correlated to effort. It can also be applied without prior change or tuning for different data sets containing different attributes. This yields the advantage of avoiding explicit knowledge elicitation.

Through evaluation of five datasets we demonstrated that early effort estimates can be significantly improved through using GFNSE model. Four out of five data sets showed in general statistically significant improvements in prediction accuracy when using GFNSE whilst the small data set (Albrecht) showed no significant difference, so it seems that any model can generally provide good predictions. Although it must be remembered that the GFNSE can be optimised on the appropriate choose of number of clusters, the results obtained have shown that careful selection of early stage attributes has considerable impact on the prediction accuracy of GFNSE. The criteria used to select attributes for early stage estimation were: (1) practical relevance for early stage estimation in that all readily available size measures (such as Function Points) were chosen and attributes like KLOC were ignored. (2) Ignoring categorical attributes at this stage as it is difficult to represent them as Fuzzy numbers using Fuzzy C-means.

Further, from the results, GFNSE would seem to throw up number of interesting advantages. First, GFNSE remains viable when ignoring non early attributes as KLOC (e.g. Kemerer, Albrecht and COCOMO datasets). Second, GFNSE remains accurate for small data sets (e. g. the Albrecht (24 projects), Kemerer (15 projects)), and for large data sets as well (e.g. ISBSG and Desharnais). Third, GFNSE remains accurate where the number of attributes is limited (e. g. the Kemerer data set).

This analysis also permits an empirical evaluation of a number of questions relating to the most effective use of GFNSE: First, "What is the optimum number of clusters for GFNSE to search for? Figure 15 shows the impact of number of clusters to use in developing GFNSE (with one analogy) for each data set. The most striking attribute of this analysis is the similarity of the five lines. For ISBSG data set, the prediction accuracy reaches a poor level of accuracy on 2 Fuzzy clusters, the trend is again a steady increase in accuracy after using 4 clusters. The general trend for Desharnais data set is less turbulent than its ISBSG. Starting off with a poor level of accuracy (The *MMRE* being over 45%) until reaching 7 clusters, GFNSE slowly improves and reaches a consistent level, at approximately between 22% and 27%. The line of COCOMO data set is approximately similar to that of Desharnais except with a bit of turbulent at cluster number 5 where suddenly the *MMRE* decreased again to 48%. In contrast, the general trend of Kemerer data set shows inconsistent behaviour, specifically at 12 clusters and after 13 clusters. Similarly, line of Albrecht data set is far less consistent but the remarkable point is the stability of *MMRE* after 7 clusters and remains markedly consistent between 50% and 58%.

Overall Findings show that there is a tendency for the *MMRE* level to improve as the number of clusters increase. Exception to this can be seen in both Kemerer and Albrecht that GFNSE backs to generate bad prediction after certain point. The principal reason behind that is due to the size of both data sets which is relatively small. For all data sets, there is a point at which the level of accuracy begins to stabilise, which indicates that estimation by GFNSE is less favourable below this number of clusters. However, the criteria by which the cut off point in the suitable number of clusters to be used is judged, remains unclear even with using *XB* formula.

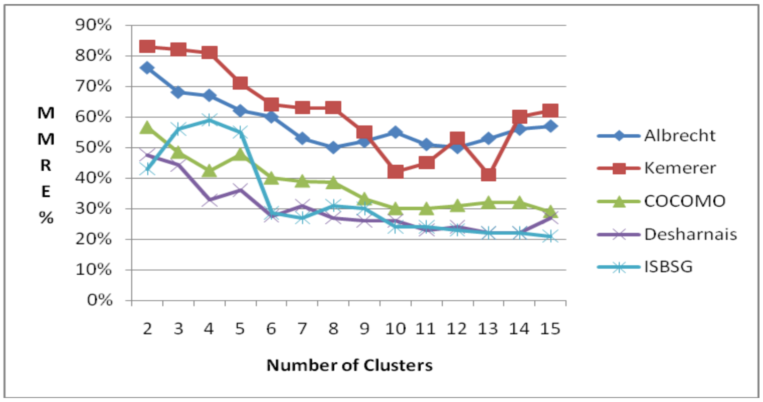

Figure 15 The investigation into the sensitivity of GFNSE to number of Clusters

The second raised question is "What is the optimum number of analogies for GFNSE to search for? the answer to the question appear to be subjective in that 'three Analogies' is the most commonly accurate estimation method for large data sets (e.g. ISBSG and Desharnais), being selected for 4 out of the 5 datasets. 'Two Analogies' was the most accurate only for Albrecht data set. The main assumptions made in previous studies (Mendes et al., 2003a; 2003b) was that the selection of just one analogy would be suitable choice for large data sets while a smaller data set would favour the selection of more analogies. This however, has not been ascertained in the results, with for example, the two largest data sets, ISBSG and Desharnais finding respectively three analogies to be the optimum number to search for. Even though the selection of 'Two Analogies' seems to be good for Albrecht dataset which is considered somehow small in size, it must be remembered that for many of the data sets the use of three different analogies methods returned remarkably predictive accuracy levels. Particularly, these findings demonstrate the effectiveness of the proposed adaptation technique.

Perhaps the main finding that can be drawn on this point is that, the larger the data set, the more consistent the results are likely to be. This point needs further investigations and will be looked at again from a different view point in the future works, where individual data sets will be examined to see if accuracy improves as data points are added. Further empirical investigation is also necessary to ensure the validity of proposed approach on other datasets and in the presence of attribute subset selection.

9. Conclusions

The inherited uncertainty at early stage of software development process increases the challenge in providing good estimates. Software effort estimation has to deal with such uncertainty and this part of research has shown advantages in explicitly modelling this uncertainty. Our research produced a software effort estimation model based on the concept of Fuzzy numbers to improve estimation accuracy. The proposed model shows it effectiveness in terms of using all available numerical attributes without the need for attribute selection. In this study, the variables selected for this kind of prediction are the same as the variables selected for predicting effort at any phase of software development process, except for some cases where the use of non-early size attribute (such as Line of Code) is not appropriate, in addition to excluding categorical attribute as they are not supported in GFNSE.

The study presented in this paper is particularly important for analogy at early stage of software development, because this topic has not been the subject of analogy research in the past. From the results of these studies on the analyzed data sets, the authors conclude that GFNSE is able to support and handle uncertainty of early stage effort estimation with using Fuzzy numbers. Although no attribute subset selection algorithm or attribute weighting technique have been used, GFNSE seems providing better prediction than the CBR and stepwise regression on the value of MMRE, confirmed by Mann-Whitney U test that showed that there is significant difference between the accuracy of the predictions of GFNSE, CBR and stepwise regression models.

In the examples with different data sets, we investigated the impact of number of analogies on the prediction accuracy. Typically, we found 2 or 3 analogies are enough to achieve best accuracy. The results also support the claim that the proposed adaptation technique based on Fuzzy numbers is good alternative to other used adaptation techniques such as linear size adjustment, mean, median and inverse rank sum. It provides uncertainty modelling by using mathematical operations of Fuzzy numbers to ensure its resilience to early stage estimation.

In this paper, we have also demonstrated that a major problem for using Fuzzy number is how to determine the spread (distribution) of a Fuzzy number. Although, there is no reliable technique that can derive the spread for a Fuzzy number, the expert opinion is still widely used, but it sometimes leads to large inconsistent results. Therefore we propose a model to derive the spreads of Fuzzy numbers based on the distribution of attribute values instead of relying on expert opinion which is often unavailable or unreliable. Our findings suggest that a good strategy for increasing estimation accuracy is to expand the use of Fuzzy logic and its relevant concepts to deal with software estimation. In this study, we have only evaluated our approach to derive spreads of Fuzzy numbers. However, we believe that expert opinion can also be integrated into Fuzzy number generation to enhance our approach where this opinion is available. In future, the best way to assess uncertainty is to let project managers determine the interval of uncertainty for each measurement during data collection.

10. Acknowledgements

The authors would like to thank ISBSG as well as the Promise Repository for granting us permission to utilize their datasets in this research.

11. References

Auer, S., Biffl, M., 2004. Increasing the Accuracy and Reliability of Analogy-Based Effort Estimation with Extensive Project Attribute Dimension Weighting, In: Proceedings of the International Symposium on Empirical Software Engineering (ISESE'04), pp. 147-155.

Azzeh, M., Neagu, D., Cowling, P., 2008a. Fuzzy Attribute subset Selection for Software Effort Estimation, In: International workshop on software predictors PROMISE'08 (part of ICSE'08), Leipzig, Germany, pp.71-78.

Azzeh, M., Neagu, D., Cowling, P., 2008b. Software project similarity measurement based on Fuzzy c-Means, In: International Conference on software process, Leipzig, Germany, pp. 123-134, 2008b.

Azzeh, M., Neagu, D., Cowling, P., 2009. Fuzzy grey relational analysis for software effort estimation, Journal of Empirical software engineering DOI. 10.1007/s10664-009-9113-0.

Bezdek, J.C., 1981. Pattern Recognition with Fuzzy Objective Function Algorithms, Kluwer Academic Publishers, Norwell, MA, New York.

Boehm B., Valerdi R., 2006. Achievements and Challenges in Software Resource Estimation, In: Proceedings of International conference on software engineering (ICSE '06), Shanghai, China.

Boetticher, G., Menzies, T., Ostrand, T., 2008. PROMISE Repository of empirical software engineering data http://promisedata.org/ repository, West Virginia University, Department of Computer Science.

Chen, S. H., 1985. Operations on Fuzzy numbers with function principal, Tamkang J. Manag. Sci. 6, 13–25.

Chen, S.J., Chen, S.M., 2003. Fuzzy risk analysis based on similarity measures of generalized Fuzzy numbers, IEEE Transactions on Fuzzy Systems 11, 45–56.

Chen S-J, Chen S-M, 2008. Fuzzy risk analysis based on measures of similarity between interval-valued Fuzzy numbers, Journal of computers & mathematics with applications 55, 1670-1685.

Chen, S. M., 1996. New Methods for subjective mental workload assessment and Fuzzy risk analysis, Cybernetics and Systems 27, 449-472.

Chiu, N.-H., Huang, S.-J., 2007. The adjusted analogy-based software effort estimation based on similarity distances. Journal of Systems and Software 80, 628-640.

Dubois, D., Prade, H., 1978. Operations on Fuzzy numbers, The International Journal of Systems Sciences 9, 613–626.

Foss T., Stensrud, E., Kitchenham, B., Myrtveit, I., 2003. A simulation Study of the Model evaluation Criterion MMRE. IEEE transaction on software engineering, 29, 985-995.

Hsieh C.H., Chen, S.H., 1999. Similarity of generalized Fuzzy numbers with graded mean integration representation, In: Proceedings of the Eighth International Fuzzy Systems Association World Congress, Taipei, Taiwan, Republic of China, pp. 551–555, 1999.

Idri, A., Abran, A., Khoshgoftaar, T., 2001. Fuzzy Analogy: a New Approach for Software Effort Estimation, In: 11th International Workshop in Software Measurements, pp. 93-101.

ISBSG, 2007. International Software Benchmarking standards Group, Data repository release 10, Site: http://www.isbsg.org.

Jain, R., 1976. Decision-making in the presence of Fuzzy variables, IEEE Transactions on Systems, Man and Cybernetics 6, 698–703.

Jain, R., 1978. A procedure for multi-aspect decision making using Fuzzy sets, The International Journal of Systems Sciences 8, 1–7.

Jorgensen M, 2004. Realism in Assessment of Effort Estimation Uncertainty: It Matter How you ask, IEEE transaction on software engineering 30, 209-217.

Jorgensen, M., Indahl, U., Sjoberg, D., 2003. Software effort estimation by analogy and "regression toward the mean", Journal of Systems and Software 68, 253-262.

Jorgensen, M., Molokken-Ostvold K., 2006. How large are software cost overruns? A review of the 1994 CHAOS report, Journal of Information and Software Technology 48, 297-301.

Jowers, L., Buckley, J., Reilly, K., 2007. Simulating continuous Fuzzy systems, Journal of information sciences 177, 436-448.

Keung, J. , Kitchenham, B., 2008. Experiments with Analogy-X for software cost estimation, In: 19th Australian Conference on software engineering, pp. 229-238, 2008.

Kitchenham, B. Linkman, S, 1997. uncertainty, Estimates, and risk, IEEE Software 69-74.

Kitchenham, B. Pickard L.M., MacDonall S. G., Shepperd M. J., 2001, What accuracy statistics really measure, IEEE proceedings-Software, 148(3), pp. 81-85.

Lee H.S., 1999. An optimal aggregation method for Fuzzy opinions of group decision, In: Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, Tokyo, Japan, pp. 314–319, 1999.

Lee, L-W, Chen S-M, 2008. Fuzzy risk analysis based on Fuzzy numbers with different shapes and different deviation, Journal of Expert Systems with Applications 34, 2763-2771.

Mendes, E., Mosley, N., Counsell, S., 2003a. A replicated assessment of the use of adaptation rules to improve Web cost estimation, In: International Symposium on Empirical Software Engineering, pp. 100-109, 2003.

Mendes, E., Mosley, N., Counsell, S., 2003b. Do adaptation rules improve web effort estimation?, In:Proceedings of the fourteenth ACM conference on Hypertext and hypermedia (Nottingham, UK), pp. 173-183, 2003.

Mendes E, Watson I, Triggs C, Mosley N, Counsell S., 2003c. A comparative study of Cost Estimation models for web hypermedia applications, *Journal of Empirical Software Engineering* 8:163-193.

Musflek, P., Pedrycz, W., Succi, G., Reforment, M., 2000. Software cost estimation with Fuzzy models. Applied Computing Review 8, 24-29.

Pfleeger, S. L., Wu, F. & Lewis, R.(2005), Software cost estimation and sizing methods: issues, and guidelines, RAND corporation.

Shepperd, M. J., Schofield, C., 1997. Estimating Software Project Effort Using Analogies, IEEE Transaction on Software Engineering 23,736-743.

Song, Q., Shepperd, M., Mair, C., 2005. Using Grey Relational Analysis to Predict Software Effort with Small Data Sets, In: Proceedings of the 11th International Symposium on Software Metrics (METRICS'05), pp. 35-45.

Wei S-J, Chen S-M, 2009a. A new approach for Fuzzy risk analysis based on similarity measures of generalized Fuzzy number, Journal of Expert Systems with Applications 36, 589-598.

Wei S-J, Chen S-M, 2009b. Fuzzy risk analysis based on interval-valued Fuzzy numbers, Journal of Expert Systems with Applications 36, 2285-2299.

X. L Xie and G. Beni, 1991. A validity measure for Fuzzy clustering."IEEE Transactions on Pattern Analysis Machine

Intelligence 13, 841-847.

Xu, Z., Khoshgoftaar, T., 2004. Identification of Fuzzy models of software effort estimation, Journal of Fuzzy Sets and Systems 145, 141-163.

Yen, K., Ghoshary, S., Roig, G., 1997. A linear regression model using triangular Fuzzy number coefficient, Journal of Fuzzy set and systems 106, 167-177.