Grey Relational Effort Analysis Technique Using Regression Methods for Software Estimation

Geeta Nagpal[1], Moin Uddin[2], and Arvinder Kaur[3]
[1]Department of Computer Science and Engineering, National Institute of Technology, India
[2]Faculty of Engineering and Technology, Delhi Technological University, India
[3]Department of Information Technology, University School of Information Technology, India

Abstract: *Software project planning and estimation is the most important confront for software developers and researchers. It incorporates estimating the size of the software project to be produced, estimating the effort required, developing initial project schedules, and ultimately, estimating on the whole cost of the project. Numerous empirical explorations have been performed on the existing methods, but they lack convergence in choosing the best prediction methodology. Analogy based estimation is still one of the most extensively used method in industry which is based on finding effort from similar projects from the project repository. Two alternative approaches using analogy for estimation have been proposed in this study. Firstly, a precise and comprehensible predictive model based on the integration of Grey Relational Analysis (GRA) and regression has been discussed. Second approach deals with the uncertainty in the software projects, and how fuzzy set theory in fusion with grey relational analysis can minimize this uncertainty. Empirical results attained are remarkable indicating that the methodologies have a great potential and can be used as a candidate approaches for software effort estimation. The results obtained using both the methods are subjected to rigorous statistical testing using Wilcoxon signed rank test.*

Keywords: *Software estimations, estimation by analogy, fuzzy clustering, robust regression, GRA.*

Received April 9, 2012; accepted March 19, 2013;published online February 26, 2014

1. Introduction

Software project development is a creative process where each person's efficiency is different. It is difficult to plan and estimate at the beginning as most software projects have deficient information and vague associations amongst effort drivers and the required effort. Software developers and researchers are using different techniques and are more concerned about accurately predicting the effort of the software product being developed. Estimation by Analogy (EbA) appears to be well suited to effort estimation, especially when the software product is poorly understood. It is concerned with finding solutions for a new problem based on known solutions from a set of similar projects. The motivation behind estimation by analogy is that the information retrieved from comparable software projects can help the management to improve the planning process, get superior in accurate bidding and for risk analysis and also similar projects tend to have similar costs. It uses a distance measure, in order to determine the similarity between two projects. EbA methods range from machine learning methods, regression techniques, and Grey Relational Analysis (GRA) a technique of Grey System Theory (GST), soft computing methods to a combination of these. In this research work grey relational analysis, a technique of GST has been applied as a similarity metrics between two projects in integration to regression and fuzzy logic.

GST a recently developed system engineering theory first established by Deng [5] draws out valuable information by generating and developing the partially known information. It has been applied in different areas of image processing [14], mobile communication [29], machine vision inspection [15], decision making [18], stock price prediction [35], and system control [9]. The accomplishment of GST motivated us to examine its application in software effort estimation. In this study, GRA a technique of GST utilizes the concept of absolute point-to-point distance between cases [30]. In the first method, the focus is effort prediction using k nearest projects from total of n projects to the reference project based on their Grey Relational Grade's (GRG), and then regressing the effort of those k projects. In the second method, the grey relational coefficient uses Fuzzy C-Means (FCM) algorithm to calculate the distance between two projects. Researchers have used various techniques from time to time for efficiently generating software effort estimates. Mukhopadhyay *et al.* [19] developed ESTOR, a CBR tool to estimate project effort. The metrics used by ESTOR are function point components and inputs to the intermediate COCOMO model. Shepperd *et al.* [27], expresses EbA in an automated environment known as ANaloGy softwarE tooL (ANGEL) that supports the collection, storage and identification of the most analogous projects from the repository in order to estimate the cost and effort. It uses Euclidean distance as the distance measure to reduce the amount of computation involved. The research was carried out on six different datasets and it has outperformed the traditional algorithmic methods. Shepperd and Schofield [28] also validated nine different industrial datasets and concluded that in all cases analogy outperforms algorithmic models based on Step wise regression. Song *et al.* [30] proposed GRA based on software ProjeCt Effort Prediction (GRACE). Huang *et al.* [13] made software effort estimation based on similarity distances. They applied genetic algorithm to analogy based software effort estimation models. It is used to derive linear model from the similarity distances between pairs of projects for adjusting the reused effort. Li *et al.* [17] describes a new flexible method called AQUA which combines the key features from two known analogy based estimation techniques: Case Based Reasoning (CBR) and Collaborative Filtering (CF). The results have demonstrated better accuracy and broader applicability by combining techniques of CBR and CF with existing analogy-based effort estimation methods. Hsu and Huang [10] use weighted grey relational analysis for software development and have proposed six weighted methods to be integrated into GRA. Jorgenson and Shepperd [16] made a very systematic review, they considered 304 studies describing research on Software Cost Estimations. Mittas and Angelis [23] compared cost prediction models by re sampling techniques. They proposed the effect of iterated bagging on EbA and validated it using artificial and real data sets. Mittas *et al.* [22] also improved analogy based cost estimation by re sampling method. Mittas and Angelis [24] combined regression and estimation by analogy in a semi-parametric model for software cost estimation. The results were improved by the utilization of this semi parametric model. Azzeh *et al.* [2, 3] have used EbA based on the integration of fuzzy set theory with GRA. In order to improve the performance of analogy based estimation at early stages of software development, two new methods based on integration of GRA with regression and fuzzy have been proposed. In the first method, GRA has been applied in order to generate the grey relational grades for the objective projects with respect to the reference project. The projects are ranked and selected on the basis of grades, effort is estimated by regressing the effort on the other independent variables of k projects. The value of k varies with each reference project. In the second method, the grey relational coefficient of GRA uses FCM algorithm to generate the distance between two projects. Both the methods have shown comparable results and also considerable improvement over GRACE [30], GRACE$^+$ [31] and FGRA [2].

The remainder of the paper is organized as follows: In section 2, a brief review of GRA, fuzzy logic and various regression techniques is provided. Section 3, describes the two proposed software effort prediction mechanisms using GREAT_RM and Fuzzy Grey Relational Analysis (FuzzyGRA), section 4, gives a brief discussion about the validation and evaluation criteria the datasets used and the empirical results obtained from evaluation of the proposed methodologies. Section 5, presents the conclusion and directions for future work.

2. Modeling Methods

2.1. Grey System Theory

GST works on unascertained systems with partially known and partially unknown information. Systems with completely unknown information are black systems. Systems with complete information available are called white systems. The term "Grey" lies between "Black" and "White" and it indicates that the information is partially available. GRA is one of the several aspects of the GST.

2.1.1. Grey Relational Analysis

GRA is comparatively a novel technique in Software Estimations for analyzing the relationships that exists between two series. The magnetism of GRA to software effort estimation shoots from its suppleness to model complex nonlinear relationship between effort and cost drivers [30]. The basic concepts of GRA is the factor space and grey relational space.

- *Factor Space*: Let $p(X)$ be a theme characterized by a factor set X, and Q be an influence relation, $\{p(X); Q\}$ is a factor space. The factor space $\{p(X); Q\}$ have the following properties: Existence of key factors, number of factors is limited and countable, factor independence, factor expansibility.
- *Comparable Series*: Suppose $x_i = \{x_i(1), x_i(2), ..., x_i(m)\}$, where $i = 0, 1, 2, ..., n \in N$; $m \in N$, is a data series. This series is said to be comparable if, and only if it is Dimensionless, Scaled and Polarized.
- *Grey Relational Space*: If all the series in a factor space $\{p(X); Q\}$ are comparable, the factor space is a grey relational space which is denoted as $\{p(X); \Gamma\}$. In a grey relational space $\{p(X); \Gamma\}$, X is a collection of data series x_i ($i = 0, 1, ..., n$), in which $x_i = \{x_i(1), x_i(2), ..., x_i(k)\}$, is the series; and $k = 1, 2, ..., m$, are the factors. Γ, which is the grey relational map set and based on geometrical mathematics, has the four properties: Normality, symmetry, entirety, and proximity.

2.1.2. Grey Relational Analysis by Deng's Method

GRA is used to quantify all the influences of various factors and the relationship among data series that is a collection of measurements [5, 6, 7].

- *Data Processing*: Data processing reduces the randomization and increase the regularity of data by using upper-bound effectiveness (i. e., larger-the-better) [8]:

$$x_i^*(k) = \frac{x_i(k) - min_i x_i(k)}{max_i x_i(k) - min_i x_i(k)} \quad (1)$$

Where $i = 1, 2, ..., m$ and $k = 1, 2, ..., n$. where $x_i(k)$ is the value of the k_{th} attribute in the i_{th} series; $x_i(k)$ is the normalized value of the k_{th} attribute in the i_{th} series; $max_i x_i(k)$ and $min_i x_i(k)$ are the maximum and minimum of the k_{th} attribute in all series.

- *Difference Series*: GRA uses the grey relational coefficient γ to describe the trend relationship between an objective series and a reference series at a given point in a system:

$$\gamma\left(x_0(k), x_i(k)\right) = \frac{\Delta_{min} + \zeta\Delta_{max}}{\Delta_{o,i}(k) + \zeta\Delta_{max}} \quad (2)$$

Where $\Delta_{0,i}(k) = |x_0(k) - x_i(k)|$ is the difference of the absolute value between $x_0(k)$ and $x_i(k)$; $\Delta_{min} = min_j min_k\ |x_0(k) - x_j(k)|$ is the smallest value of $\Delta_{0,j}$ $\forall j \in \{1, 2, ..., n\}$; $\Delta_{max} = max_j max_k\ |x_0(k) - x_j(k)|$ is the largest value of $\Delta_{0,j}$ $\forall j \in \{1, 2, ..., n\}$; and ζ is the distinguishing coefficient, $\zeta \in (0, 1]$. The ζ value will change the magnitude of $\gamma(x_0(k), x_i(k))$. In this study the value of ζ has been taken as 0.5 [2].

- *Grey Relational Grade*: GRG is used to find overall similarity degree between reference tuple x_o and comparative tuple x_i. When the value of GRG approaches 1, the two tuples are "more closely similar". When GRG approaches a value 0, the two tuples are "more dissimilar". The GRG $\Gamma(x_0, x_i)$ between an objective series x_i and the reference series x_0 was defined by Deng as follows:

$$\Gamma\left(x_0, x_i\right) = \frac{1}{n}\sum_{k=1}^{n}\gamma\left(x_0(k), x_i(k)\right) \quad (3)$$

2.2. Fuzzy Set Theory

Fuzzy set theory was introduced by Zadeh [37] and provides mathematical capabilities to handle ambiguous or vague concepts of human perception for complex systems problems, where it is extremely difficult to build the system models mathematically. It presents a structure to associate fuzzy sets to linguistic values. Every set is symbolized by Triangle, Trapezoidal, Gaussian, Sigmoid etc., and assigns a membership value between 0 and 1 for each point in the universe of disclosure. The membership value represents how much a point belongs to the fuzzy set [36]. Fuzzy models are generally used for simulation, identification of system behavior as well as for the prediction and control purposes.

2.3. Regression Techniques

Regression analysis is a statistical technique for modelling and analysis of variables. It is used to study the relationship that exists between dependent variable and one or more independent variables.

2.3.1. Ordinary Least Square Regression

It is the most popular and widely applied technique to build software cost estimation models. According to principle of least squares the 'best fitting' line is the line which minimizes the deviations of the observed data away from the line. It is referred to as multiple linear regressions and is given by:

$$y_i = \beta_0 + \beta_1 x_{i,1} + ...\beta_k x_{i,k} + \varepsilon_i \quad (4)$$

Where, Y_i is a dependent variable where as $x_1, x_2, ..., x_k$ are k independent variables. β_o is the y intercept, β_1, β_2 are the slope of y, ε_i is the error term. The corresponding prediction equation is given as:

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_{i,1} + ... + \hat{\beta}_k x_{i,k} \quad (5)$$

In this equation $\hat{\beta}_0, \hat{\beta}_1, ..., \hat{\beta}_k$ are the least square coefficients and $\hat{y}_i$ is estimated response for i^{th} term. Thus, the response estimated from the regression line minimizes the sum of squared distances between the regression line and the observed response. The least square method tries to minimize $\sum e_i^2$.

2.3.2. Robust Regression and M Estimation

It is a type of regression technique which prevails over limitations of OLS. OLS estimates are extremely non-robust to outliers. Outliers should be either treated or detached from analysis as they can inefficiently influence the whole process of fitting. Robust Regression (RR) [1, 32, 42] is an application of Iteratively Reweighted Least Squares (IRLS) regression in which the weights are iteratively set by taking the residual terms of the previous iteration into consideration The least square method tries to minimize $\sum e_i^2$, which is unstable in case there are outliers present in the data, whereas the RR M Estimators tries to minimize the effect of these outliers by minimizing $\sum w_i^2 e_i^2$ in each iteration. The steps involved in the Iteratively Reweighted Least Square (ILRS) [20] are:

- *Step 1*: *In the first iteration*, each observation is allocated equal weight and the coefficients of the model are estimated using Ordinary Least Squares (OLS).
- *Step 2*: *In the second step*, after the OLS, residuals are used to find weights. The observation with larger residual is assigned lower weight.
- *Step 3*: *In the third iteration*, the new model parameters and the residuals are recomputed using Weighted Least Squares (WLS).
- *Step 4*: *In step 4*, new weights as per step 2 are found and the procedure continues until the values of the parameter estimates converge within a specified tolerance. The tuning constants play a significant role in the performance of estimators as they establish the shape and cutoff points of

weighting functions. The objective function, weight function and tuning constant [11, 12, 21] for all the estimators is shown in Table 1.

Table 1. Estimators with weight functions.

Estimator	Objective function $\rho_H(e)$	Weight function $w_H(e)$	Tuning Constant
Bisquare	$\begin{cases} \frac{k^2}{6}\left\{1-\left[1-\left(\frac{e}{k}\right)^2\right]^3\right\} & for \lvert e\rvert \le k \\ \frac{k^2}{6} & for \lvert e\rvert > k \end{cases}$	$\begin{cases} \left[1-\left(\frac{e}{k}\right)^2\right]^2 & for \lvert e\rvert \le k \\ 0 & for \lvert e\rvert > k \end{cases}$	4.685
Fair	$k^2\left(\frac{\lvert e\rvert}{k}-\log\left(1+\frac{\lvert e\rvert}{k}\right)\right)$	$\left(1+\frac{\lvert e\rvert}{k}\right)^{-1}$	1.400
Huber	$\begin{cases} \frac{1}{2}e^2 & for \lvert e\rvert \le k \\ k\lvert e\rvert-\frac{1}{2}k^2 & for \lvert e\rvert > k \end{cases}$	$\begin{cases} 1 & for \lvert e\rvert \le k \\ \frac{k}{\lvert e\rvert} & for \lvert e\rvert > k \end{cases}$	1.345

2.3.3. Stepwise Regression

It is method for adding and removing terms based on their statistical importance. The forward approach starts with no variables in the model, trying out the variables one by one and including them if they are 'statistically important'. The selection has been used for estimating the effort of reference project from various similar projects. At each step, a predictor is entered based on partial *F*-tests or *t*-test. The procedure continues till more variables can be justifiably entered. The first variable that is put in the stepwise model is the variable having the smallest *t*-test *P*-value (below α_E = 0.05). The level of significance (α) is taken to be 5%.

3. Proposed Methodologies

3.1. Modeling Grey Relational Effort Analysis Technique with Regression Methods

Modeling Grey Relational Effort Analysis Technique with Regression Methods (GREAT_RM) focuses on project selection based on GRA and effort prediction by regression. In the GRA based studies so far, effort is estimated by generating similar projects to the target project and then estimating effort from the *k* most similar projects. This methodology, uses GRA for generating similar projects but effort is generated by applying regression on best *k* projects most similar to the target project. The value of *k* varies with each reference project. The structural framework of GREAT_RM is shown in Figure 1.

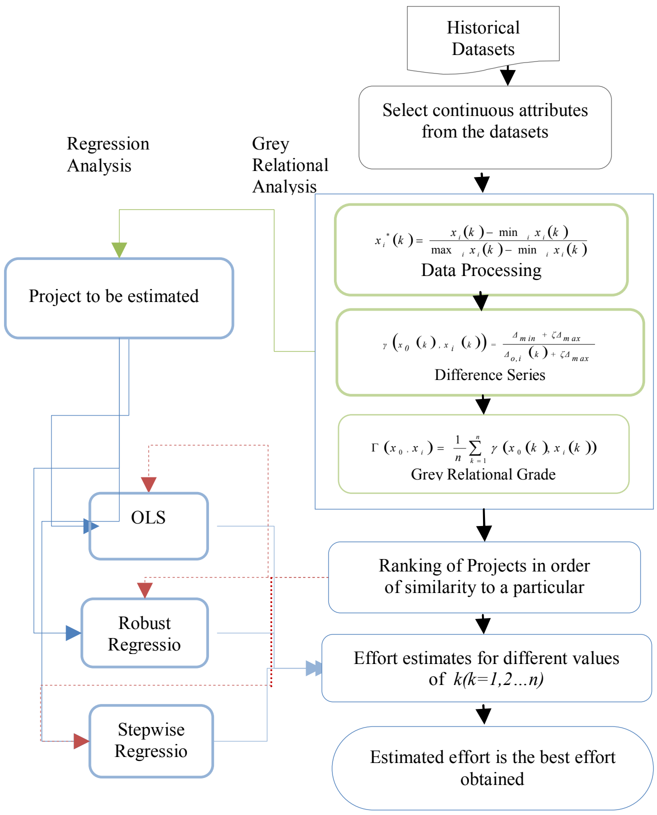


Figure 1. Structural framework of GREAT_ RM technique.

The basic steps of the methodology are:

- *Step 1*: Select continuous attributes from the dataset.
- *Step 2*: *Data Series Construction,* the data set consists of series $x_0 = \{x_1(1), x_1(2), \ldots, x_1(m)\}$, $x_1 = \{x_2(1), x_2(2), \ldots, x_2(m)\}$, $x_2=\{x_3(1), x_3(2), \ldots, x_3(m)\}$ and $x_n=\{x_n(1), x_n(2), \ldots, x_n(m)\}$, x_0 is the reference series whose effort is to be estimated based on the objective series $x_1, x_2, \ldots x_n$.
- *Step 3*: *Data Preparation*, the numerical features are normalized in a specified range so that each feature has same weight on effort.
- *Step 4*: *Ranking k Closest Projects,* this aims at retrieving software projects from the dataset that exhibit large similarity with project under investigation. The distance between two tuples at k^{th} feature, is calculated by the formula as shown in Equation 6:

$$\gamma(x_0(k), x_i(k)) = \frac{\Delta_{min}+\zeta\Delta_{max}}{\Delta_{0,i}(k)+\zeta\Delta_{max}} \tag{6}$$

All GRG between the reference project *o* and i^{th} comparative project, $\Gamma\,(x_o, x_i)$ are calculated according to Equation 6. The range of Γ is from 0 to 1 in each case. For more similarity between projects the value of Γ approaches one and for zero it means that two projects are completely dissimilar, the projects that have the higher value on GRG gets the greatest opportunity to contribute in the final estimate.

- *Step 5*: *Effort Prediction by GRA*, effort for GRA is the simple aggregation of k most influential projects [25]:

$$\hat{\varepsilon} = \sum_{i=1}^{k} w_i * \varepsilon_i \tag{7}$$

Where weight w_i is given by:

$$w_i = \frac{\tau(x_0, x_i)}{\sum_{j=1}^{k} \tau(x_0, x_i)} \tag{8}$$

ε_i = effort of i^{th} most influential project.

- *Step 6*: *Effort Prediction by Regression*, in this step, effort estimate for a given project is calculated by applying various regression techniques on only *k* most similar projects obtained from step 2.

3.2. Modeling using Fuzzy Grey Relational Analysis for Software Effort Estimation

The framework of Fuzzy GRA for individual projects is shown in Figure 2. The steps involved in the process are as explained below:

- *Step 1*: Select continuous attributes from the dataset.
- *Step 2*: Data Series Construction, (explained in step 2, section 3.1).
- *Step 3*: Data Preparation, normalization of data and list wise deletion of missing values are performed in the data preparation step.
- *Step 4*: Case Retrieval, in this step, those projects are retrieved from the dataset that exhibit large similarity with the reference project. $\Delta_{0,i}(k)$ between two tuples is calculated using the fuzzy distance, the data is fuzzified using FCM before calculating the distance (explained in clustering data). After generating the grey relational coefficient between the tuples, the grey relational grades between the reference project *o* and i^{th} comparative project are calculated, the $\Gamma(x_o, x_i)$ values are calculated for each *i* according to Equation 3. The projects are ranked on the basis of their grades and *k* projects with highest grades are selected. The value of *k* is different with each reference project.

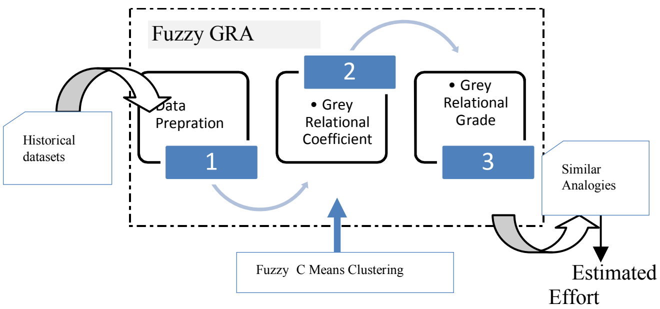


Figure 2. Structural framework for Fuzzy GRA software effort estimation.

3.2.1. Clustering Data

To the data values obtained after normalization, we apply FCM clustering, for generating the $\Delta_{0,i}(k)$ of the grey relational coefficient. Instead of generating the Δ as per Equation 2, we make significant modification to $\Delta_{0,i}(k)$ by incorporating the fuzzy distance between two numeric feature values. The purpose behind using fuzzy is that it tries to minimize the uncertainty associated with similarity measurement.

FCM algorithm enables to group closest projects together in the same cluster enabling better and efficient case retrieval. It clusters the data set into n clusters with every data point in the dataset belonging to every cluster to a certain degree. *genfis3* a function of MATLAB, generates a FIS using FCM clustering by extracting a set of rules that models the data behavior. This algorithm is based on the minimizing the objective function that signifies the distance from any given data point to a cluster center weighted by that data point's membership grade:

fismat = genfis3(Xin, Xout, type,cluster_n*)* (9)

The function requires separate sets of input and output data as input arguments. It generates a FIS structure and allows specifying the number of clusters (cluster_n) to be generated by FCM. The number of clusters determines the number of rules and membership functions in the generated FIS, cluster_n is assigned either an integer or 'auto'. In this study cluster_n is 'auto', the function uses the subtractive clustering algorithm with a radii of 0.5 and the minimum and maximum values of X_{in} and X_{out} as xBounds to find the number of clusters.

The subtractive clustering method assumes each data point as a potential cluster center and calculates a measure of the likelihood that each data point would define the cluster center, based on the density of surrounding data points. The steps involved are:

- Selects the data point with the highest potential to be the first cluster center
- Removes all data points in the vicinity of the first cluster center (as determined by radii), in order to determine the next data cluster and its center location
- Iterates on this process until all of the data is within radii of a cluster center

4. Experimental methods

4.1. Validation and Evaluation Criteria

The datasets available have less than 100 observations, therefore instead of using hold out or 10 fold validation, Leave one Out Cross Validation (LOOCV) has been applied. In LOOCV, for each iteration one project is left out once as test data and used entirely to assess the performance of the data set that is trained on the remaining projects. In order to measure the accuracy of the software estimation, we have used four most popularly used evaluation criteria.

Table 2. Evaluation criteria's.

Performance Measures	Formula
Magnitude of Relative Error (MRE)	$MRE = \frac{\lvert actual_i - estimated_i \rvert}{actual_i}$
Mean MRE(MMRE)	$MMRE = \frac{1}{N}\sum_{X=1}^{N} MRE_x$
(Median of MRE's) MdMRE	$MdMRE = median\ (MRE_x)$
Magnitude of Relative Error relative to the estimate (MER)	$MER = \frac{\lvert actual_i - estimated_i \rvert}{estimated_i}$
Mean MER(MMER)	$MMER = \frac{1}{N}\sum_{X=1}^{N} MER_x$
Pred(l)	k/n*100

The boxplots of the absolute residuals $|actual_i$-$estimated_i|$ gives good indication of the distribution of residuals and can explain summary statistics such as MMRE and Pred (25). The Box plot shows the five number summary i. e., the median as the central tendency of the distributions, the Inter Quartile Range (IQR) and the min-max values. It also shows the outliers of the individual distributions. The length is the spread of the distribution. The box represents 50% of the observations in the distribution. A small box is a peaked distribution, whereas a long box is flattened distribution.

4.2. Data Sources

In order to evaluate the models based upon the prosposed methodologies, five well established datasets from the Promise repository [26] have been used for validating our models. The descriptive statistics of the data sets are shown in Table 3 given below. Though these dataset are old, still they are extensively being used to assess the comparative accuracy of the new technique.

Table 3. Descriptive Statistics of the datasets.

	Dataset	Cases	Features	Effort Mean	Effort Standard Deviation
1	Finnish	38	8	7678.29	7135.28(hours)
2	Desharnais	77	9	4834	4188(hours)
3	Cocomo-81	63	17	683.526	1821.51133 (hours)
4	Albrecht	24	8	21875	28417(hours)
5	Kemerer	15	5	219.25	263(man hours)

5. Experimental Results

The experimental results of the proposed models are shown in the Table 9.

5.1. Comparison for Finnish Dataset

The best results have been achieved with Finnish dataset, with MMRE = 10.03% and Pred (25) = 89.47%. With Fair robust estimator integrated with GRA. Using FuzzyGRA, also equally good results were obtained with MMRE = 21.66 and Pred (25) accuracy = 81.57%.

5.2. Comparison for Albrecht Dataset

As shown in Table 4. GRACE model that was developed by Song *et al.* [30], obtained MMRE = 60.25% and Pred (25) = 52.63%. In the second study by Song *et al.* [31] obtained MMRE = 26.1% and Pred (25) = 50%. In the study carried out by Shepperd and Schofield [27] between regression and analogy estimation models on Albrecht dataset, regression model resulted in MMRE = 90% and Pred = 33%, while analogy obtained MMRE = 62% and Pred = 33%. The FGRA model by Azzeh [2] obtained MMRE = 51.1% and Pred (25) = 48.6%. The GREAT_RM model appears significantly better than those of Shepperd results, GRACE results and FGRA results with MMRE = 24.16% and Pred (25) = 70.83%. Best results were obtained using FuzzyGRA demonstrating that fuzzy performs the best while handling uncertainties.

Table 4. Comparison over albrecht dataset.

ALBRECHT				
	MMRE	Md MRE	Pred(25)	MMER
Fuzzy GRA (individual projects)	45.01	3.96	75.00	16.33
GREAT_RM	24.16	10	70.83	22.57
GRACE+	26.1	24.2	50	-
FGRA	51.1	48	28.6	60.4
GRACE	60.25	21.35	52.63	-

5.2. Comparison for COCOMO-81 Dataset

The GREAT_RM results have outperformed the other techniques in case of COCOMO-81 dataset. Song *et al.* [30] applied the GRACE model to the COCOMO dataset and they obtained MMRE = 76.09%, and Pred (25) = 20.63%. Another study by Song *et al.* [31] obtained MMRE = 49.8% and Pred (25) = 29%. FGRA [2] obtained MMRE = 23.2% and Pred (25) = 66.7%. GREAT_RM results obtained MMRE = 21.04% and Pred (25) = 76.19% which is an improvement over GRACE FGRA and GRACE$^+$.

Table 5. Comparison over cocomo_81 dataset.

Cocomo 81				
	MMRE	Md MRE	Pred(25)	MMER
GREAT_RM	21.04	9.42	76.19	48.71
GRACE+	49.8	55.2	29	-
FGRA	23.2	14.8	66.7	25.6
GRACE	76.09	60.52	20.63	-

5.3. Comparison for Desharnais Dataset

The Desharnais dataset has been widely used to test software estimation models. Shepperd and Schofield [27] employed analogy estimation on 77 available projects after removing four projects that have missing values using Angel tool. They obtained MMRE = 64%

and Pred (25) = 36%. In similar way, Song *et al.* [30] developed GRACE software estimation model based on grey relational analysis, they obtained MMRE = 49.83%, Pred (25) = 30%. FGRA obtained MMRE = 30.6% and Pred (25) = 64.7% . Another study by Song *et al.* [31], GRACE$^+$ obtained MMRE = 41.4%, Pred (25) = 45.3%. Thus *GREAT_RM* achieves the best MMRE = 19.48 % and Pred (25) = 89.61%. Equally good results were obtained using FuzzyGRA with MMRE = 24.6% and Pred (25) = 71.42%.

Table 6. Comparison over desharnais.

Desharnais				
	MMRE	**MdMRE**	**Pred(25)**	**MMER**
Fuzzy GRA (individual projects)	24.69	7.40	71.42	23.59
GREAT_RM	16.78	7.84	74.02	31.63
GRACE+	41.4	29.2	45.3	-
FGRA	30.6	17.5	64.7	34.4
GRACE	49.83	33.93	30	-

5.3.1. Comparison for Kemerer Dataset

For the Kemerer dataset, the fair robust estimator gives better results when used with GRA. FuzzyGRA also produced reasonable results with MMRE = 29.63% and Pred (25) = 60.00%. GRACE obtained MMRE = 58.83% and Pred (25) = 26.67%. Another study GRACE$^+$ by Song *et al.* (2011) obtained MMRE = 19.6% and Pred (25) = 78.6%. The FGRA model by Azzeh [2] obtained MMRE = 36.2% and Pred (25) = 52.9%. The GREAT_RM model appears significantly better than those to GRACE results and FGRA results, however less significant to GRACE $^+$.

Table 7. Comparison over kemerer dataset.

Kemerer				
	MMRE	**Md MRE**	**Pred(25)**	**MMER**
Fuzzy GRA (individual projects)	41.19	8.58	66.67	33.21
GREAT_RM	29.63	16.56	60	27.93
GRACE+	19.6	13.8	78.6	-
FGRA	36.2	33.2	52.9	34.3
GRACE	58.83	46.94	26.67	-

The boxplots of the residuals obtained suggest that:

- The medians are very close to zero, signifying that the estimates were biased towards the minimum value where they have tighter spread.
- The median and range of absolute residuals of both methods are small, that shows that at least half of the predictions are accurate.
- The boxes of GRA, GRA+OLS, GRA+RR and Fuzzy GRA overlays the lower tail for all datasets which also presents accurate predictions.
- The results of GRA+SWR were not very accurate, this can be observed from the boxplots also.

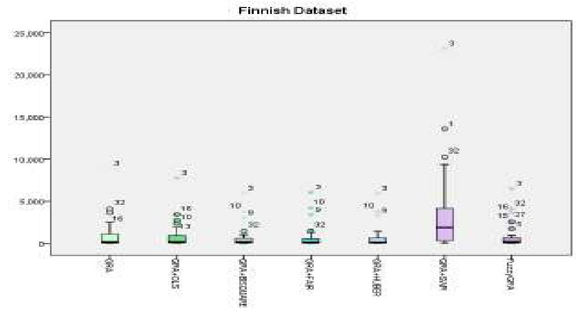


Figure 3. Box plot of absolute residuals for finnish dataset.

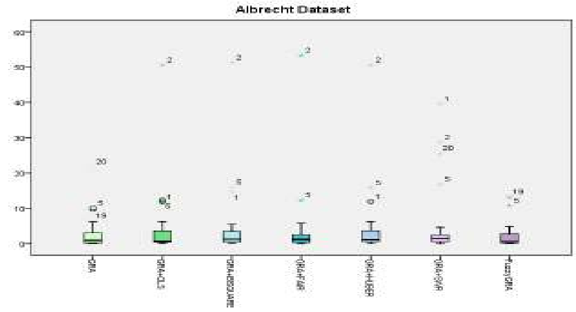


Figure 4. Box plot of absolute residuals for albrecht dataset.

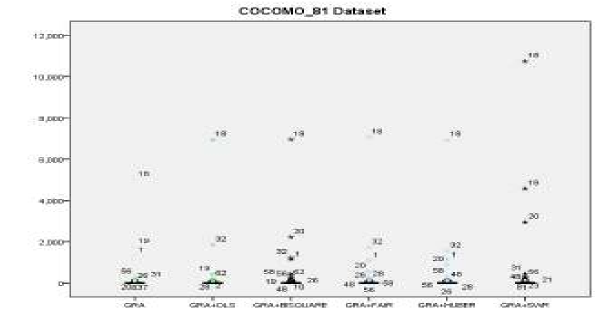


Figure 5. Box plot of absolute residuals for cocomo-81 dataset.

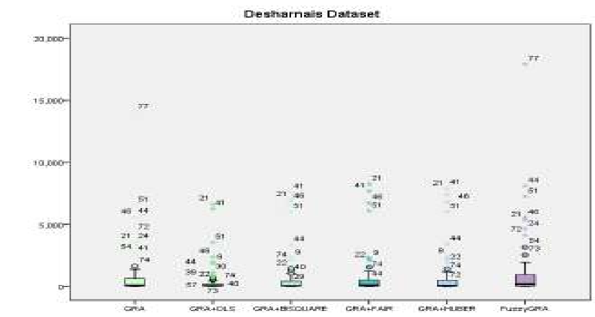


Figure 6. Box plot of absolute residuals for desharnais dataset.

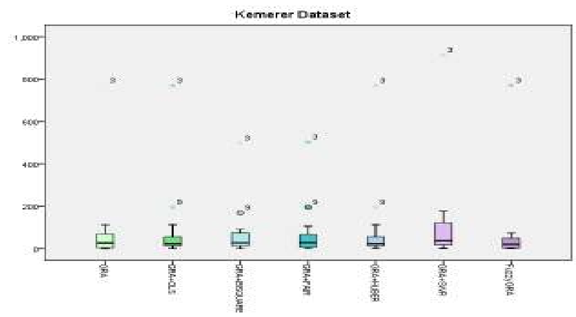


Figure 7. Box plot of absolute residuals for kemerer dataset.

Table 8. Wilcoxon signed rank test.

	GRA - Actual	(GRA+OLS) - Actual	(GRA+Bisquare) - Actual	(GRA+Fair) - Actual	(GRA+Huber) - Actual	(GRA+SWR) - Actual	FuzzyGRA - Actual)
Z (Finnish)	-.138[a]	-.964[a]	-.457[a]	-.732[a]	-.703[a]	-1.922[b]	.000[c]
Asymp. Sig. (2-tailed)	0.89	0.335	.648	.464	.482	0.055	1.000
Z(Albrecht)	-.729[a]	-.171[a]	-.457[b]	-.629[a]	-.514[b]	-.743[a]	-.800[a]
Asymp. Sig. (2-tailed)	0.466	0.864	.648	.530	0.607	0.458	.424
Z(COCOMO_81)	-.698[a]	-1.595[a]	-1.171[a]	-1.458[a]	-1.184[b]	-2.615[b]	
Asymp. Sig. (2-tailed)	0.485	0.111	.242	.145	.236	.009	
Z(Desharnais)	-1.234[a]	-.784[a]	-.724[b]	-.145[b]	-.307[b]	-4.359[b]	-.129[b]
Asymp. Sig. (2-tailed)	0.217	0.433	.469	.885	.759	0	.897
Z(Kemerer)	-.369[a]	-.568[b]	.000[a]	.000[c]	-.568[b]	-2.385[b]	-.483[a]
Asymp. Sig. (2-tailed)	0.712	0.57	1	1.000	.570	0.017	.629
a. Based on negative ranks. b. Based on positive ranks. c. Wilcoxon signed ranks test							

Table 9. Outcomes of the two methodologies (GRA, GRA+OLS, GRA+ Bisquare, GRA+ Fair, GRA+ Huber, GRA+SWR, Fuzzy GRA).

DATA SET	GRA	GRA + OLS	GRA + Bisquare	GRA+ Fair	GRA+ Huber	GRA + Stepwise	Fuzzy GRA
Finnish							
MMRE	11.37	11.88	11.41	10.3	10.76	58.61	21.66
Median (MRE)	2.67	2.19	2.3	1.93	2.07	27.93	3.30
MMER	12	12.88	7.81	9.81	10.52	40.99	11.59
Pred(25)	76.32	89.47	89.47	89.47	92.1	47.37	81.57
Albrecht							
MMRE	46.35	29.83	27.09	26.15	24.16	32.64	45.01
Median (MRE)	4.89	7.47	11.31	7.72	10	12.39	3.96
MMER	17.15	21.72	24.43	21.62	22.57	24.87	16.33
Pred(25)	70.83	70.83	70.83	66.67	70.83	70.83	75.00
COCOMO -81							
MMRE	30	32.35	37.37	24.66	28.52	21.04	--
Median (MRE)	7.07	5.32	5.59	5.45	3.86	9.42	--
MMER	26.86	15.71	19.91	19.72	21.42	48.71	--
Pred(25)	68.25	76.19	71.43	73.02	74.6	76.19	--
Desharnais							
MMRE	34.9	18.19	25.44	28.07	28.21	16.78	24.69
Median (MRE)	5.07	1.51	8.36	8.44	7.31	7.84	7.40
MMER	22.12	8.74	18.97	19.42	19.23	31.63	23.59
Pred(25)	68.83	90.9	79.22	75.32	79.22	74.02	71.42
Kemerer							
MMRE	46.67	35.18	32.68	29.63	35.12	38.65	41.19
Median (MRE)	11.46	21.92	28.37	16.56	29.1	31.35	8.58
MMER	35.58	41.04	29.88	27.93	30.61	78.8	33.21
Pred(25)	60	53.33	46.67	60	46.67	40	66.67

The results achieved by GREAT_RM and FuzzyGRA are subjected to statistical testing using Wilcoxon signed rank test by setting the test value to 0 shown in Table 8. The level of significance is taken to be 0.05. If the resulting p-value is less than 0.05, then a statistically significant difference exists between the sample median and test value but in case it is greater than 0.05 than it can accepted that no statistical difference exists between sample median and test value.

- Most of the results of the residuals obtained in most of the cases were not significantly different from the test value and hence the null hypotheses were accepted in the case of GRA, GRA with OLS, GRA with RR and FuzzyGRA for all datasets.
- The results obtained with integration of GRA and SWR for Desharnais, Kemerer and COCOMO-81 dataset were however significantly different and cannot be accepted.

6. Conclusions and Future Scope

The empirical evaluations have revealed that the GREAT_RM and FuzzyGRA techniques can certainly improve the estimation process. The models can be used for early stage estimation where the data is uncertain. The methodology presents a significant improvement over GRACE [30] and FGRA [2]. The results obtained are superior over our previous results wherein the value of k was fixed for each reference project using GREAT_RM [25].

Thus, the results obtained are inspiring and urge us to endeavour different methodologies for producing enhanced estimates. The proposed methodologies can further be applied on some other large datasets with different validation criteria's and also feature selection methodology can be applied in both methods to enhance prediction. Further, GREAT_RM methodology can be carried out with other Robust Regression techniques like S-Estimators, Least Trimmed Squares or MM Estimations etc.

Acknowledgment

The authors would like to thank Promise Repository for providing such valuable datasets for carrying out research work.

References

[1] Andrews D., Bickel P., Hampel F., Huber P., Rogers W., and Tukey J., *Robust Estimates of Location: Survey and Advances*, Princeton University Press, USA, 1972.

[2] Azzeh M., Neagu D., and Cowling P., "Fuzzy Grey Relational Analysis for Software Effort Estimation," *Journal of Empirical Software Engineering*, vol. 15, no. 1, pp. 60 - 90, 2010.

[3] Azzeh M., Neagu D., and Cowling P., "Analogy-based Software Effort Estimation Using Fuzzy Numbers," *Journal of Systems and Software*, vol. 84, no. 2, pp. 270 - 284, 2011.

[4] Boehm B. and Valerdi R., "Achievements and Challenges in Software Resources Estimation," *IEEE Software*, vol. 25, no. 5, pp. 74 - 83, 2008.

[5] Deng J., "Control Problems of Grey System," *System and Control Letters*, vol. 1, no. 5, pp. 288 -94, 1982.

[6] Deng J., "Introduction to Grey System Theory," *the Journal of Grey System*, vol. 1, no. 1, pp. 1-24, 1989.

[7] Deng J., "Grey Information Space," *The Journal of Grey System*, vol. 1, no. 1, pp. 103 - 117,1989.

[8] Hsia K. and Wu J., "A Study on the Data Preprocessing in Grey Relation Analysis," *Journal of Chinese Grey System Association*, vol. 1, no. 1, pp. 47 - 53, 1998.

[9] Hsu C. and Huang C., "Improving Effort Estimation Accuracy by Weighted Grey Relational Analysis During Software Development," *in Proceedings of the 14th Asia-Pacific Software Engineering Conference*, Washington, USA, pp. 534 - 541, 2007

[10] Huang S., Chiu N., and Chen L., "Integration of the Grey Relational Analysis with Genetic Algorithm for Software Effort Estimation," *European Journal of Operational and Research* vol. 188, no. 3, pp. 898 - 909, 2007.

[11] Huang S. and Huang C., "Control of an Inverted Pendulum using Grey Prediction Model," *IEEE Transactions on Industry Applications*, vol. 36, no. 2, pp. 452 - 458, 2000.

[12] Huber P., "Robust Estimation of a Location Parameter," *Annals of Mathematical Statistics*, vol. 35, no. 1, pp. 73 - 101, 1964.

[13] Huber P., "Robust Regression: Asymptotics, Conjectures and Monte Carlo," *The Annals of Statistics*, vol. 1, pp. 799 - 821,1981.

[14] Jiang B., Tasi S., and Wang C., "Machine Vision-Based Gray Relational Theory Applied to IC Marking Inspection," *IEEE Transactions on Semiconductor Manufacturing*, vol. 15, no. 4, pp. 531 - 539, 2002.

[15] Jou J., Chen P., and Sun J., "The Gray Prediction Search Algorithm for Block Motion Estimation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 6, pp. 843 - 848, 1999.

[16] Jorgenson J. and Shepperd M., "A Systematic Review of Software Development Cost Estimation Studies," *IEEE Transactions on Software Engineering*, vol. 33, no. 1, pp. 33 - 53, 2007.

[17] Li G., Ruhe J., Emran A., and Richter M., "A Flexible method for software Effort Estimation by Analogy," *Empirical Software Engineering*, vol. 12, no. 1, pp. 65 - 106, 2007.

[18] Luo R., Chen T., and Su K., "Target Tracking Using a Hierarchical Grey-Fuzzy Motion Decision Making Method," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 31, no. 3, pp. 179 - 186, 2001.

[19] MATLAB® Documentation, avalable at: http://www.math works.com/help/techdoc/, last visited 1994.

[20] Mbamalu G. and Hawary M., "Load Forecasting via Suboptimal Seasonal Autoregressive models and Iteratively Reweighted Least Squares Estimation" *IEEE Transactions on Power Systems*, vol. 8, no. 1, pp. 343 - 347, 1993.

[21] Mittas N. and Angelis L., "Comparing Cost Prediction models by Resampling Techniques," *Journal of Systems and Software*, vol. 81, no. 5, pp. 616 - 632, 2008.

[22] Mittas N., Athanasiades M., and Angelis L., "Improving Analogy-based Software Cost Estimation by a Resampling Method," *Journal of Information and Software Technology*, vol. 50, no. 3, pp. 221 - 230, 2008.

[23] Mittas N. and Angelis L., "LSEbA: Least Squares Regression and Estimation by Analogy in a Semi-Parametric Model for Software Cost Estimation," *Empirical Software Engineering*, vol. 15, no. 5, pp. 523 - 555, 2010.

[24] Mukhopadhyay T., Vicinanza S., and Prietula M., "Examining the Feasibility of a Case-Based Reasoning Model for Software Effort Estimation," *MIS Quarterly*, vol. 16, no. 2, pp. 155 - 171, 1992.

[25] Nagpal G., Uddin M., and Kaur A., "A Hybrid Technique Using Grey Relational Analysis and Regression for Software Effort Estimation Using Feature Selection," *the International Journal of Soft Computing and Engineering*, vol. 1, no. 6, pp. 20 - 27, 2012.

[26] PROMISE Repository of Empirical Software Engineering Data, http://promisedata.org/repository.

[27] Shepperd M., Schofield C., and Kitchenham B., "Effort Estimation Using Analogy," *in Proceedings of the 18th International Conference on Software Engineering*, Berlin, Germany, pp. 170 - 178, 1996.

[28] Shepperd M. and Schofield C., "Estimating Software Project Effort Using Analogies," *IEEE Transaction on Software Engineering*, vol. 23, no. 11, pp. 736 - 743, 1997.

[29] Song Q., Shepperd M., and Mair C.,"Using Grey Relational Analysis to Predict Software Effort with Small Data Sets," *Proceedings of the 11th International Symposium on Software Metrics*, Como, Italy, pp. 35 - 45, 2005.

[30] Song Q. and Shepperd M., "Predicting Software Project Effort: A Grey Relational Analysis Based Method," *Expert Systems with Applications*, vol. 38, no. 6, pp. 7302 - 7316, 2011.

[31] Su S., Su Y., and Huang J., "Grey-Based Power Control for DS-CDMA Cellular Mobile Systems," *IEEE Transactions on Vehicular Technology*, vol. 49, no. 6, pp. 2081 - 2088, 2000.

[32] Verardi V. and Croux C., "Robust Regression in Stata," *Stata Journal, StataCorp LP*, vol. 9, no. 3, pp. 439 - 453, 2009.

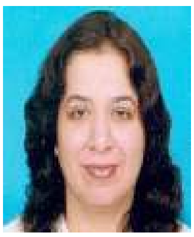
[33] Wang Y., "On-Demand Forecasting of Stock prices Using a Real-Time Predictor," *IEEE Transactions on Knowledge and Data Engineering*, vol. 15, no. 4, pp. 1033 - 1037, 2003.

[34] Wu J., You M., and Wen K., "A Modified Grey Relational Analysis," *The Journal of Grey System*, vol. 11, no. 3, pp. 287 - 292, 1999.

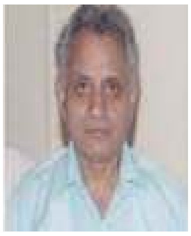
[35] Wu J. and Chen C.,"An Alternative form For Grey Relational Grades," *the Journal of Grey System*, vol. 11, no. 1, pp. 7 - 12, 1999.

[36] Yan L., "Modeling Fuzzy Data with Fuzzy Data Types in Fuzzy Database and XML Models," *the International Arab Journal of Information Technology*, vol. 10, no. 6, pp. 610 - 615, 2013.

[37] Zadeh L., "Toward a Theory of Fuzzy Information Granulation and its Centrality in Human Reasoning and Fuzzy Logic," *Journal Fuzzy Sets and Systems*, vol. 90, no. 2, pp. 111 - 127, 1997.



Geeta Nagpal pursuing PhD programme in the Department of Computer Science and Engineering, National Institute of Technology, Jalandhar, INDIA. She completed her Master's degree in Computer Science from Punjab Agricultural University, Ludhiana. She is presently working as Associate Professor in the Department of Computer Science and Engineering at NIT, Jalandhar. Her research interests are Software Engineering, Databases and Data mining.



Moin Uddin Pro Vice Chancellor, DTU, Delhi, India. He obtained his BSc Engineering and M.Sc. Engineering (Electrical) from AMU, Aligarh in 1972 and 1978 respectively. He obtained hid Ph. D degree from University of Roorkee, Roorkee in 1994. Before Joining as the Pro Vice Chancellor of Delhi Technological University, he was the Director of NIT, Jalandhar for five years. He has worked as Head Electrical Engineering Department and Dean Faculty of Engineering and Technology at Jamia Millia Islamia (Central University) NewDelhi. He supervised 14 Ph. D thesis and more than 30 M.Tech dissertations.

Arvinder Kaur Associate Professor, University School of IT, GGSIPU, Delhi, India. She completed her Master's degree in Computer Science from Thapar Institute of Engineering and Technology and Ph D from Guru Gobind Singh Indraprastha University, Delhi. Prior to joining the school, she worked with Dr. B. R. Ambedkar Regional Engineering College, Jalandhar and Thapar Institute of Engineering and Technology. Her research interests include Software Engineering, Object-Oriented Software Engineering, Software Metrics, Microprocessors, Operating Systems, Artificial Intelligence, and Computer networks. She is a lifetime member of ISTE and CSI. Her paper titled, "Analysis of object oriented Metrics" was published as a chapter in the book Innovations in Software Measurement (Shaker-Verlag, Aachen 2005).