Improving Effort Estimation of Fuzzy Analogy using Feature Subset Selection

Ali Idri and Safae Cherradi
Software Projects Management Research Team
University Mohamed V in Rabat, Morocco
ali.idri@um5.ac.ma, sfcherradi@gmail.com

***Abstract*—Feature selection has been recently used in the area of software development effort estimation for improving the accuracy and robustness of prediction techniques. The idea behind selecting the most informative subset of features from a pool of available effort drivers stems from the hypothesis that reducing the dimensionality of datasets may significantly minimize the complexity and time required to reach to an optimal and accurate estimation. This paper compares two relatively popular feature selection techniques (Forward Subset Selection and Backward Feature Elimination) used with Fuzzy Analogy for software effort estimation. This empirical comparison is done over eight well-known datasets with the Jackknife evaluation method. The results suggest that Fuzzy Analogy using feature subset selection generates more accurate estimates in terms of the Standardized Accuracy (SA) and Pred(p) criteria than Fuzzy Analogy without using feature subset selection regardless of the data set used. Moreover, this study found that the use of Forward Feature Selection, rather than Backward Feature Elimination, may improve the prediction accuracy of Fuzzy Analogy and reduce the number of features selected.**

Keywords- Software Development Effort Estimation, Fuzzy Analogy, Feature Subset Selection.

I. Introduction

Software Development Effort Estimation (SDEE) is a basic operation in project management processes. Essentially it includes estimating the person-months necessary for developing software. The development process usually starts after the specification of a product and continue until the implementation of a product and release. In the aim to achieve this, loads of models have been developed with the ultimate goal to have successful project completion and develop high quality software with the lowest possible costs [1].
However, effort/cost estimations still usually inaccurate because of some cost drivers that affect development effort relate to highly subjective characteristics, and are not yet adequately investigated, defined or understood by researchers [2-4]. In addition, due to the fact that software development is driven by a large number of factors that are consequently used in effort estimation models, it can result in strong co-linearity, heteroscedasticity and unstable prediction accuracy [5,6].

There are some SDEE techniques those have recently gained ground in literature namely Machine learning (ML) techniques [7]. Analogy-based Software Effort Estimation (ASEE) and Artificial Neural Networks (ANN) were the most used SDEE techniques among eight types of machine learning identified by Wen et al. [7], accounting for 37% and 26%, respectively. This confirms the results of the study [1]: ASEE is gaining in popularity over other ML techniques (10% compared to 7% for ANN and 5% for Classification and Regression Trees up to 2004).

Idri et al. [8] conducted a systematic literature review (SLR) in which 65 relevant papers were selected to gain more knowledge about the use of ASEE. Their results suggest that ASEE provides better accuracy (49.8% for Mean Magnitude Relative Error, 29.4% for Median of MREs, and 51.2% for Pred(25)). Moreover, ASEE outperforms the other prediction models and can model the complex relationships between effort and software attributes. Furthermore, ASEE improves estimation accuracy when it is used in combination with other techniques specifically fuzzy logic, genetic algorithms, the model tree and collaborative filtering. However, their SLR showed that ASEE still have some serious challenges regardless of data sets characteristics:

- It cannot adequately handle categorical data other than binary valued features[9];
- it cannot be directly applied to historical data sets containing missing data [2]; and
- it is sensitive to irrelevant features and the degree of feature influence on effort estimates [3].

Recall that the ASEE process is usually composed of three steps: Feature and Case Subset Selection (FCSS), similarity evaluation (SE), and adaptation (AD). The FCSS step deals especially with feature selection among other issues such as case selection, missing values and categorical data. Hence, FSS techniques are often investigated in the FCSS step. The principal findings of the review of Idri et al. [8] concerning the FCSS step showed that: 1) The FCSS step was performed in the most of the selected studies (63%), followed by the Adaptation step (57%), and, finally, the Similarity evaluation step (34%); 2) There was a significant interest on dealing with missing values and category attributes rather than FSS techniques in the FCSS step; and 3) Fuzzy Logic (FL) and Statistical Methods(SM) were the most frequently used techniques in the FCSS step, followed by Genetic Algorithm (GA).

However, owing to the insufficient number of studies evaluating the impact of FSS techniques used in combination with ASEE methods, the review of Idri et al. [8] recommends that these results need to be investigated in further research. Therefore, this paper is concerned with the use of two feature subset selection techniques with the Fuzzy Analogy effort estimation technique presented in Section IV.

Previous investigations emphasized on the ability of feature subset selection techniques on improving the accuracy of ASEE. Azzeh et al. [10] examined the impact of Hill climbing, Exhaustive Search, Forward/Backward Feature Selection and Fuzzy Feature Subset Selection on ASEE and confirmed that FSS has a significant impact on accuracy of ASEE. Also, Li et al. [11] affirmed that FSS techniques gave better results when using Mutual Information, Hill Climbing, Exhaustive Search and Forward Feature Selection with ASEE. Regarding the impact of FSS techniques, Azzeh et al. [12] found that the use of Forward Feature Selection yields mostly the best results for all ASEE techniques.

This study investigates the influence of two FSS methods: Forward Feature Selection (FFS) and Backward Feature Elimination (BFE) on Fuzzy Analogy prediction performance. The aim is to assess the appropriateness of these two FSS techniques when using Fuzzy Analogy, which will subsequently lead to a better understanding of the features that may be used for achieving high estimation performance, or to considering their relevance, redundancy or irrelevancy for Fuzzy Analogy. To do that, this study evaluates the accuracy of Fuzzy Analogy with FFS and BFE on eight historical software project datasets. Therefore, two research questions were discussed:

RQ1: Does the use of FFS and BFE improves prediction accuracy of Fuzzy Analogy?

RQ2: Which project features should we use as inputs for Fuzzy Analogy?

The rest of this paper is organized as follows: Section II presents the feature selection approaches FFS and BFE used in this study and Section III gives an overview of related research work investigating FSS techniques in ASEE. Section IV presents an overview of Fuzzy Analogy. Section V presents the experiment design including data sets description, criteria for evaluating estimation accuracy, and experiment process. The results obtained are presented and discussed in Section VI. Section VII summarizes the conclusions and suggests further research.

II. FEATURE SUBSET SELECTION

The fundamental aim of feature subset selection approaches is the identification of the most significant features subset for a specific problem [13, 14]. In general, feature selection approaches may belong to one of three categories: Filters, Wrappers, or Embedded algorithms [15].

Filter techniques assess the relevance of features by looking only at the intrinsic properties of the data. In most cases a feature relevance score is calculated, and low-scoring features are removed. Afterwards, this subset of features is presented as input to a classification/prediction algorithm. In Filtering methods, a feature can be selected based upon some predefined criteria such as Mutual Information [16,17], Independent Components Analysis [18] and Correlation-based feature selection [19].

Whereas Filter techniques treat the problem of finding an optimal feature subset independently of the classification/prediction model selection step, Wrapper methods embed the model hypothesis search within the feature subset search. In this setup, a search procedure in the space of possible feature subsets is defined, and various subsets of features are generated and evaluated. The evaluation of a specific subset of features is obtained by training and testing a specific classification/prediction model, rendering this approach tailored to this model. To search the space of all feature subsets, a search algorithm is then 'wrapped' around the classification/prediction model. Examples of Wrapper techniques are FFS, BFE [48] and GA [20].

In a third category of FSS techniques, termed Embedded techniques, the search for an optimal subset of features is built into the classifier construction, and can be seen as a search in the combined space of feature subsets and hypotheses. Just like Wrapper approaches, Embedded approaches are thus specific to a given classification/prediction model. Examples of Embedded techniques are Decision Trees Weighted naïve Bayes and Feature selection using the weight vector of SVM [21,22,23].

In this work, two popular FSS techniques (Forward and Backward Feature Selections (FFS and BFE) have been used and as Wrappers with Fuzzy Analogy for software effort estimation due to their simplicity and their ease of adaptation to various ML techniques [11,24-27]. As shown in Fig. 1, FFS/BFE starts from an empty/full feature set and iteratively adds/removes a certain feature a_k whose addition/removal to the current subset results in the highest prediction accuracy. This iterative process ends when the addition/removal of any other feature does not result in an improvement [28]. FFS/BFE is a popular technique that has proven to yield good results for ASEE: Azzeh et al. [10] examine the impact of FSS techniques on ASEE and they recommended FFS/BFE when the computation time is important. Moreover, Azzeh et al. [12] claimed that the use of FFS prior building ASEE techniques provides accurate estimates than those derived without using FFS. Dejaeger et al. [25] affirmed that BFE leads to a remarkable accuracy improvement of ASEE techniques and other SDEE models as well.

III. RELATED WORK

Many researchers questioned the necessity of a large number of features involved usually in SDEE techniques and moreover, investigations showed that in most cases redundant features could be eliminated [29,23]. Feature selection involves finding the optimum subset of features that provides the most accurate prediction. Some earlier research work involved correlations for dimensionality reduction [30], Exhaustive Search of features [31], Hill Climbing and Forward Sequential Selection [28]. Menzies et al. [32] proposed that FSS should be regularly carried out in SDEE.

To the best of our knowledge, there are relatively few studies that investigated the use of FSS techniques with ASEE. Nevertheless, the published studies on the subject have confirmed that the use of FSS techniques improves the estimation model performances.

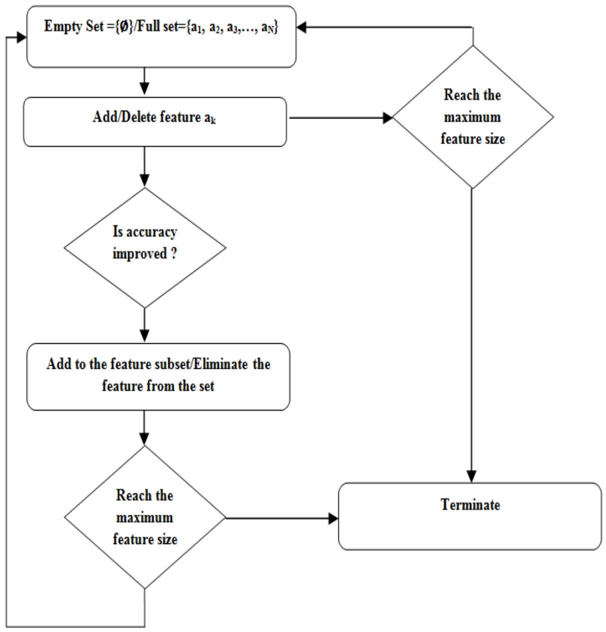


Fig. 1.FFS & BFE algorithms.

Li et al. [11] affirmed that all FSS techniques evaluated in their study, performed well and the technique MI (Mutual Information) could achieve slightly better predictions. The results show likewise that the MI can obtain more meaningful features explained by MI diagram, while wrapper selectors do not always select informative features since they merely optimize the error metric MMRE. The results of the experiments, performed by Keung et al. [33], indicate that the use of Stepwise variable selection with ASEE was very important, since the Mantel's correlation was improved using the reduced feature set. As well, the results of [34] suggested that the performance of a technique may be maintained at similar levels with only a small subset of the original features. Azzeh et al. [10] used Exhaustive Search, Hill Climbing, Random Search, FFS, BFE and Fuzzy FSS (FFSS) in combination with ASEE techniques and found that the results for the ISBSG data set were better than those obtained for Desharnais dataset.

IV. Fuzzy Analogy

This study investigates an analogy-based effort estimation technique: Fuzzy Analogy. This technique involves three steps: identification of cases, retrieval of similar cases, and case adaptation [9,35] which are described below. Each step is a "fuzzification" of the traditional procedure cost estimation by analogy. Its objective is to allow tolerance inaccuracies throughout estimation by analogy process and the management of uncertainty in the estimated costs.

A. Identification of case

In this step, each project is described by a set of relevant and independent attributes. These attributes can be measured by either numerical or linguistic values. Unlike traditional ASEE approaches in which numerical values are represented by classical intervals, in Fuzzy Analogy numerical values are transformed into linguistic ones. Let us suppose that a project P is described by M numerical/linguistic variables (V_j). Then, for each numerical/linguistic variable V_j, a measure with linguistic values is defined (A^j_k). Each linguistic value A^j_k is represented by a fuzzy set with a membership function $\mu_{A^j_k}$.

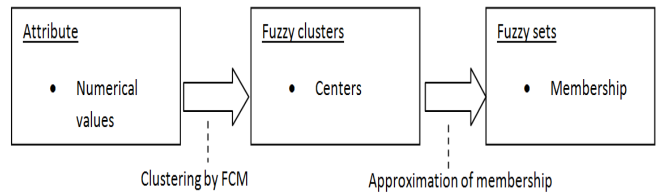


Fig. 2. Fuzzy set generation process.

These fuzzy sets and their membership functions can be built (1) empirically, using expert knowledge [35,36], or (2) automatically, using clustering techniques [37,38]. In particular, when the descriptions of software attributes are insufficient to empirically build their fuzzy representations, Fuzzy Analogy uses an automated process to build fuzzy sets and their membership functions [37,38]. The proposed fuzzy set generation process is based on the fuzzy C-means clustering technique (FCM) and a real coded genetic algorithm (RCGA). This process consists of two main steps, as shown in Fig.2. First, the well-known FCM algorithm and the Xie–Beni validity criterion are used to decide on the number of clusters (fuzzy sets) [39,40].For each software project attribute, several experiments are conducted with the FCM algorithm with different number of clusters (c) and different values of the parameter m. For each attribute, we choose the number of clusters and the parameter m that minimize the value of the Xie–Beni criterion [37]. Second, an RCGA is used to build membership functions for these fuzzy sets [41,42]. Membership functions can be trapezoidal, triangular, or Gaussian.

B. Retrieval of similar cases

To measure the similarity between two software projects described by linguistic values such as 'low' or 'high', Fuzzy Analogy proposes a set of new measures based on fuzzy logic [43]. Two steps are required for evaluating similarity using these measures: /

Individual similarities: Assessing the similarity between two projects P_1 and P_2, according to each individual attribute V_j describing P_1 and P_2 by means of (4).

$$S_{V_j}(P_1, P_2) = \begin{cases} \text{Max} - \text{min aggregation} \\ \text{Max} \min(\mu_{A^j_k}(P_1), \mu_{A^j_k}(P_2)) \\ \text{Sum} - \text{product aggregation} \\ \sum_k (\mu_{A^j_k}(P_1) * \mu_{A^j_k}(P_2)) \end{cases} \quad (1)$$

Global similarities: Evaluating overall similarity S (P_1,P_2) by aggregating the individual similarities S (P_1, P_2) using Regular Increasing Monotone (RIM) linguistic quantifiers, such as 'all', 'most', 'many', or 'some'. The choice of the appropriate RIM linguistic quantifier, Q, depends on the characteristics and needs of each environment. Q indicates the proportion of individual distances that we feel is necessary for a good evaluation of the overall distance. For example, if we choose the linguistic quantifier 'all', that means all the individual similarities were considered in the evaluation of S (P_1, P_2). The choice of the linguistic quantifier used in this study is explained in Section VI. The overall similarity of P_1 and P_2, S (P_1, P_2) is given by one of the formulas of (2).

$$S(P_1, P_2) = \begin{cases} \text{All of } (S_{V_j}(P_1, P_2)) \\ \text{Most of } (S_{V_j}(P_1, P_2)) \\ \text{Many of } (S_{V_j}(P_1, P_2))\ldots \\ \text{Some of } (S_{V_j}(P1, P2)) \end{cases} \quad (2)$$

C. *Case adaptation*

The objective of this step is to derive an estimate for the new project by using the known effort values of similar projects. The first task is to decide how many similar projects will be used in the adaptation phase to estimate the effort of the new project. Fuzzy Analogy has proposed a strategy based on similarity measures S (P, P_i) and the definition adopted in the studied environment for the proposition ' P_i is a project that is highly similar to P'. Intuitively, P_i is highly similar to P if S (P, P_i) is in the vicinity of 1. To represent the value 'vicinity of 1', we use a fuzzy set defined in the interval [0,1]. The second task is how to adapt the chosen analogies in order to generate an estimate for the new project. Fuzzy Analogy uses the weighted mean of all known effort projects in the dataset; the weights being the similarity distance. The formula is given by (3).

$$\text{Effort}(P) = \frac{\sum_{I=1}^{N} \mu_{\text{vicinity of } 1}(S(P,Pi)) * \text{Effort}(Pi)}{\sum_{I=1}^{N} \mu_{\text{vicinity of } 1}(S(P,Pi))} \quad (3)$$

where μ_{vincity} of 1 is the membership function representing the linguistic value 'vicinity of 1'.

V. EXPERIMENT DESIGN

This section describes the experimental process used in this study. It consists of two main steps: feature subset selection and accuracy evaluation. The study was designed to apply Fuzzy Analogy with FFS/BFE techniques on eight data sets. Both FFS and BFE techniques are implemented as Wrappers. Hence, the experimental design consisted in evaluating 2 FSS techniques × 1 ASEE technique × 8 datasets = 16 different effort estimation experiments. The results of each experiment are discussed using the estimation accuracy criteria Standardized Accuracy and Pred(25) described in Section V.

A. *Data description*

In this study, eight data sets were used: ISBSG repository (release 8), COCOMO81, Desharnais, Maxwell, Miyazaki, China, Kemerer and Albrecht. Table I provides an overview of these data sets, including number of attributes, observations, and previous use. The minimum, mean and maximum of effort and size are given. A histogram of effort is provided for each data set. Further details about the attributes description of the eight datasets may be found in PROMISE Repository [44]. Since the aim of this study is to select the relevant features more precisely numerical ones, the solution adopted was (1) to consider all the numerical attributes for the China, Miyazaki, Desharnais, Albrecht and Kemerer data sets, and (2) in the case of the ISBSG and COCOMO81 data sets, in which projects are described with more than 50 and 17 attributes (numerical and categorical) respectively, 10 and 16 numerical attributes were selected from those data sets respectively, as they have been used in our previous studies [9, 35,45].

B. *Accuracy evaluation*

In this study, the performance of Fuzzy Analogy was evaluated using: (1) Jackknife method, also called leave-one-out cross validation: the target project is excluded from the historical dataset and estimated by the rest of the historical projects; and (2) the Standardized Accuracy (SA) measure proposed by Shepperd and MacDonell [46] and the Prediction Level at level 0.25 defined by (4) and (7) respectively:

$$SA = 1 - \frac{\overline{MAE_{pi}}}{MAE_{p0}} \quad (4)$$

where MAE_{P_0} is defined as the MAE of the estimation method P_i and MAE_{P_0} is the mean of a large number of random guesses (in our case 1000). In the random guessing procedure, a training instance is randomly chosen with equal probability from the training set (with replacement) and its effort value is used as the estimate of the test instance. SA gives us an idea of how good an estimation method is in comparison to random guessing. Since the term MAE_{P_i} is in the nominator, the higher the SA values, the better the estimation method. The interpretation of SA is that the ratio represents how much better the predictive model (P_i) is than the mean or random guessing (P_0). A value close to zero is discouraging and a negative value would be worrisome. A positive value SA means the predictive model is better than mean or random guessing.

To verify if the predictions of a model are generated by chance, the effect size criterion defined by (5) was used:

$$\Delta = \frac{MAE_{P_i} - \overline{MAE}_{P0}}{S_{P0}} \quad (5)$$

where S_{P_0} is the sample standard deviation of the random guessing strategy. The values of Δ can be interpreted in terms of the categories proposed by Cohen [42] of small (≈0:2), medium (≈ 0:5) and large (≈0:8).

The second performance criterion used in this work is the Prediction at level 0.25. The Pred(25) measure is based on the Magnitude Relative Error (MRE) measure. MRE is calculated for each project in the dataset following (6):

$$MRE = \left| \frac{\text{Effort}_{\text{actual}} - \text{Effort}_{\text{estimated}}}{\text{Effort}_{\text{actual}}} \right| \times 100 \quad (6)$$

Pred(p) is defined as the percentage of successful predictions falling within p% of the actual values, and is calculated by (7):

$$\text{Pred}(p) = \frac{K}{N} \quad (7)$$

where N is the total number of observations, and k is the number of observations with an MRE less than or equal to p (p =25 in this study). In the literature, a model having a value of Pred (25) equal to 70% is said acceptable [47].

TABLE I. OVERVIEW OF DESCRIPTIVE STATISTICS OF HISTORICAL DATA SETS.

Dataset	Size	Unit	#Features	Effort					
				Min	***Max***	***Mean***	***Median***	***Skewness***	***Kurtosis***
Albrecht	24	Man/Months	7	0.5	105	22	11	2.30	4.7
COCOMO81	252	Man/Months	16	6	11400	683	98	4.5	21.9
China	499	Man/Hours	17	26	54620	3921	1829	3.92	19.3
Desharnais	77	Man/Hours	7	546	23940	4830	3542	2.03	5.3
ISBSG	148	Man/Hours	11	24	60270	6242	2461	3.05	11.3
Kemerer	15	Man/Months	6	23	1107	219	130	3.07	10.6
Miyazaki	48	Man/Months	7	5.6	1586	87	38	6.26	41.35
Maxwell	62	Man/Hours	26	583	63694	8223	5189	3.34	13.7

VI. RESULTS

This section presents the experimental results of the impact of FFS and BFE techniques on the accuracy of Fuzzy Analogy. The discussion is structured to answer the two research questions RQs 1-2 of the Section I. First, the impacts of the two FSS techniques were explored to analyze their influence on the accuracy of Fuzzy Analogy measured in terms of SA and Pred(25) criteria. Moreover, the effect size is discussed to check whether the predictions of Fuzzy Analogy are generated by chance and to justify if there is a large effect improvement over random guessing. Second, the number and the relevance of features selected by FFS and BFE were discussed in the aim to examine the performance of Fuzzy Analogy when the feature set is reduced.

We evaluated the accuracy of Fuzzy Analogy through a set of experiments using the eight data sets. The calculations were made using a software prototype developed with Matlab 7.0 in a Microsoft environment. The developed prototype allows us to apply various α -RIM linguistic quantifiers (Q) to the eight data sets. Usually, an appropriate quantifier must be defined for each environment by studying its features and its requirements. In this evaluation, various α-RIM linguistic quantifiers were used to calculate the overall project similarities since the appropriate quantifier for the environment from which the data set was collected was unknown. The α –RIM linguistic quantifiers used here are those defined by (8):

$$Q(r)=r^{\alpha} \qquad \alpha > 0 \tag{9}$$

where α is the proportion of attributes we feel is necessary to evaluate the similarity between projects. In all experiments, we used the max-min aggregation of (1) to evaluate the individual similarities. The results obtained indicated that the accuracy of the estimates is, in general, monotone increasing as a function of α. This is consistent with the findings of [9,35]:

- When α tends towards zero, the accuracy decreases because the overall similarity takes into account fewer attributes among all those describing software projects.
- For the other α-RIM linguistic quantifiers (α tends towards ∞), the accuracy increases (a high value of SA) with α, because additional attributes are considered in the evaluation of the overall similarity. The maximum number to consider is all attributes.

We therefore chose the value of α that gave the best results in terms of accuracy, which in most cases was the highest one (α= 500).

A. *RQ1: Impacts of FFS and BFE techniques on the accuracy of Fuzzy Analogy*

This subsection presents the experimental results of the impact of FFS and BFE techniques on the predictive accuracy of Fuzzy Analogy. First, the effect of the FFS and BFE techniques is analyzed and discussed using SA and Pred(25) criteria to evaluate their influence on the estimation accuracy of Fuzzy Analogy. Second, effect size is discussed to check whether the predictions of Fuzzy Analogy are generated by chance and to justify if there is an effect improvement over random guessing and to verify the bias of underestimated values of Pred(25) respectively.

1) *Accuracy of Fuzzy Analogy with FFS*

Fig 3. a-b shows SA and Pred(25) values when evaluating the accuracy of Fuzzy Analogy with the FFS technique. It can be observed that:

- According to the SA criterion, Fig. 3-a shows that Fuzzy Analogy using FFS provides higher accuracy values than Fuzzy Analogy without FFS for the Kemerer (SA = 42.82% instead of 24.30), Maxwell (SA = 62.80% instead of 42.74%), COCOMO81 (SA = 60.17 % instead of 44.87%), China (SA = 71.29% instead of 66.29%) and Desharnais (SA= 51.94% instead of 38.35%). However, it provides slightly higher accuracy for the ISBSG (SA = 63.44 instead of 60.37%) and Miyazaki (SA = 52.15% instead of 50%); while the use of FFS with Fuzzy Analogy provides slightly lower accuracy for the Albrecht dataset (SA = 77.65% instead of 83.32%).
- According to the Pred(25) criterion, Fig.3-b shows that Fuzzy Analogy with FFS provides very good results than Fuzzy Analogy without FFS for the Kemerer (Pred(25) = 46.66% instead of 20%), Maxwell (Pred(25) = 35.48% instead of 17.74%), Desharnais (Pred(25) = 40.25% instead of 31.16%), and Miyazaki (Pred(25 = 47.91% instead of 35.41%); whereas a slight difference is noticed for COCOMO81 (Pred(25) = 12.69% instead of 9.52%), China (Pred(25) = 30.86% instead of 27.45%) and ISBSG

(Pred(25) = 25.67% instead of 21.62%). For Albrecht, Fuzzy Analogy with/out FFS give the same Pred(25) values (58.33%).

2) Accuracy of Fuzzy Analogy with BFE

The results of the evaluation of Fuzzy Analogy with BFE in terms of SA and Pred(25) criteria are presented in Fig.3 a-b:

- According to the SA criterion, Fig 3-a shows that Fuzzy Analogy with BFE provides significant improvement for all the data sets: for Kemerer (SA = 39.77% instead of 24.30%), Maxwell (SA = 52.04% instead of 42.74%), COCOMO81 (SA = 60.89% instead of 44.87%), and Desharnais (SA = 47.47% instead of 38.35%). A slight improvement was recorded for China (SA = 69.21% instead of 66.29%), Miyazaki (SA = 54.21% instead of 50%) and ISBSG (SA = 62.81% instead of 60.37%). For Albrecht, Fuzzy Analogy provides the same results with/out BFE (SA = 83.34% instead of 83.32%).
- According to the Pred(25) criterion, Fig. 3-b shows that Fuzzy Analogy with BFE provides a higher accuracy in term of Pred(25): for Kemerer (Pred(25) = 40% instead of 20%), Maxwell (Pred(25) = 24.19%instead of 17.74%), COCOMO81 and Desharnais (Pred(25) = 36.36% instead of 31.16%). However, a slight improvement was obtained for China (Pred(25) = 30.26% instead of 27.45%), Miyazaki (Pred(25) = 27.08% instead of 22.91%) and ISBSG (Pred(25) = 25.67% instead of 21.62%). For Albrecht, Fuzzy Analogy provides the same results with/out BFE (Pred(25) = 58.33% instead of 58.33%).

3) Accuracy comparison of Fuzzy Analogy with FFS/BFE

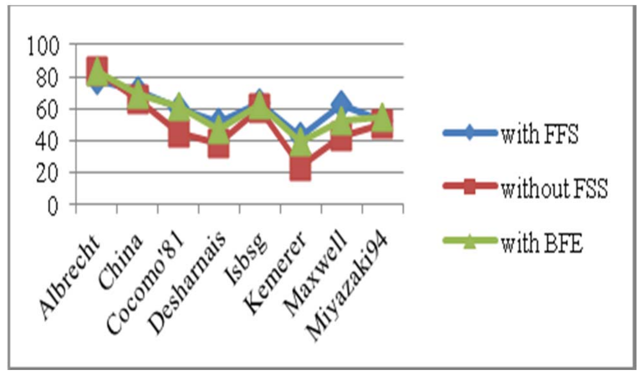
It can be seen from the above discussion that there is differences and similarities between FFS/BFE when used with Fuzzy Analogy in terms of SA end Pred(25):

- According to the SA criterion, Fuzzy Analogy with FFS generates accurate results than Fuzzy Analogy with BFE for all data sets except for COCOMO81 and Albrecht: it gives similar results in terms of SA for COCOMO81 (60%) and low value for Albrecht (FFS: SA = 77.65%; BFE: SA = 83.34%).
- According to the Pred(25) criterion, Fuzzy Analogy using FFS provides better results over Fuzzy Analogy using BFE for China, COCOMO81, ISBSG and Maxwell. However, it provides lower accuracy for Desharnais (12.69% vs 15.87%) and similar Pred(25) values for Albrecht, Kemerer and Miyazaki.

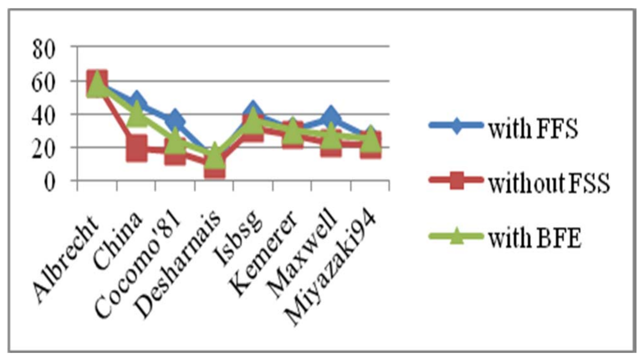
Moreover, to test whether the estimates of Fuzzy Analogy are generated by chance in order to evaluate if there is an effect improvement over random guessing, the effect size values were used.. The Δ values computed in all experiments are higher than 0.8, which means that the results obtained by SA are more likely not to be due to chance.

B. *RQ2: Which project features should we use as inputs for Fuzzy Analogy?*

Table II lists the features that were selected by FFS and BFE when used with Fuzzy Analogy for the eight data sets. As shown in Table II, BFE removed one or two features from the full set unlike FFS that provides different results. FFS selected 15% of the original features for Maxwell and 13% of COCOMO81. For the other data sets, FFS selected around the half of the features set: 41% for China, 50% for Kemerer, 43% for Albrecht, 64% for ISBSG, 43% for Desharnais and 57% for Miyazaki. Also, it presents few similarities with BFE in terms of features selected excluding for ISBSG (55% of features in common). It can be seen from the above discussion that Fuzzy Analogy with FFS achieved better results over Fuzzy Analogy with BFE while selecting features: FFS selected less features than BFE and the SA and Pred(25) accuracy are positively influenced for all data sets, except for Albrecht and Miyazaki.



(a) SA



(b) Pred(25)

Fig. 3.a-b: SA and Pred(25) values of Fuzzy Analogy with/out FFS/BFE.

Moreover, a basic objective of this study is to highlight and understand the features that are found more relevant than the rest across the experiments and the two FSS techniques. As shown in Table II, the attributes AFP, Output, Enquiry, PDR_UFP and Duration are considered important for China, by both FSS techniques. For Miyazaki, KLOC, FORM and EFILE are found the most relevant and have a considerable effect over effort. While for ISBSG, FFS and BFE selected UFP, MTS, UBL, UBCU, OC, EC and IFC as the best inputs to generate accurate effort estimation. For Desharnais, COCOMO81 and Albrecht, there are two significant attributes for each one: PointAdjust and Dev.Env, STOR and LOC, FPAdj and AdjFP respectively. The language of programming, the duration of project and the adjusted function points are the relevant attributes for Kemerer and the size of the application significantly affect the effort estimation for Maxwell.

TABLE II. SELECTED FEATURES OF EACH DATA SET WHEN USING FUZZY ANALOGY WITH FFS AND BFE.

Datasets	Number of Features	Features selected by FFS (IDs*)	Feat (%)	Features selected by BFE (IDs*)	Feat (%)	Common features selected (IDs*)	Feat (%)
China	17	17,16,11,3,2,4,1	41	All except 8,14	88	17,16,11,3,2,4,1	41
Miyazaki	7	1,2,3,7	57	All except 2,5	71	1,3,7	43
ISBSG	11	2,3,6,11,9,8,5	64	All except 1,11	82	2,3,6,9,8,5	55
Desharnais	7	6,7,5	43	2,3,4,5,7	71	5,7	29
Kemerer	6	1,3,5	50	All except 4,5	67	1,3	33
COCOMO81	16	5,16	13	All except 9,11	88	5,16	13
Albrecht	7	5,6,7	43	All except 6	86	5,7	29
Maxwell	26	25,24,17,11	15	All except 7,12	92	25,24,11	12

(*) IDs represent the identification of features for each dataset

VII. THREATS TO VALIDITY

This paper reported an evaluation of Fuzzy Analogy when using two FSS techniques. While we have attempted to design the experiments to be as comprehensive and general as possible, using different data sets, this study is still limited by many challenges.

A. *Internal validity*

Most of the internal limitations are a consequence of the scope of this study. Threats to internal validity may mainly come from the criteria used to measure the prediction accuracy. We based the findings of this study on the SA measure, which is an unbiased measure and Pred(25) which was used to provide more easily interpretable results. However, since we restricted the study to numerical attributes, there still remains the case of categorical attributes, given that most of the well-known historical software project data sets such as ISBSG and COCOMO81 contain many such attributes.

B. *External validity*

There are two threats to the external validity of this study: (1) generalization of the experiment results and (2) experiment design. The eight data sets in this study are considered randomly selected and sufficiently representative based on the following observations: (1) they were collected at different times by different organizations around the world; (2) they contain software projects in different application domains; and (3) they are of different sizes in terms of number of attributes and projects. Consequently, we assume that the number of data sets used is sufficient to enable us to draw general conclusions. Other threats may come from the FSS techniques. Although the experiments were performed using two of the most popular FSS techniques (FFS and BFE), it still compulsory to test other FSS techniques with Fuzzy Analogy.

VIII. CONCLUSION AND FUTURE WORK

This paper evaluated the use of Fuzzy Analogy technique in combination with two FFS techniques to evolve the selection of cost drivers used as inputs for the estimation of software effort. The findings of the comparison of Fuzzy Analogy when using either FFS or BFE in terms of prediction accuracy (SA and Pred(25) and the number of features suggest that: 1) Fuzzy Analogy using FSS technique generally generates better or the same accuracy than Fuzzy Analogy without FFS; 2) Fuzzy Analogy using FFS provides better SA and Pred(25) values than Fuzzy Analogy using BFE; 3) Fuzzy Analogy with FFS/BFE outperformed random guessing regardless the data set; and 4) FFS selected less than the half of features especially for large datasets and the features selected by the two FSS techniques are different for almost all the data sets.

Further work will investigate the use of Fuzzy Analogy in combination with other FFS techniques such as Greedy Hill Climbing, Exhaustive Search and Genetic Algorithm. Moreover, the results presented in this work are only preliminary, since we investigated only numerical data. Consequently, further investigation may be required on mixed data (numerical and categorical) to confirm our findings.

ACKNOWLEDGMENT

This work was conducted within the research project MPHR-PPR1-2015-2018. The authors would like to thank the Moroccan MESRSFC and CNRST for their support.

REFERENCES

[1] M. Jørgensen and MJ. Shepperd. “A Systematic Review of Software Development Cost Estimation Studies”, IEEE Transactions on Software Engineering,vol. 33 , pp.33-53, January 2007.

[2] LC. Briand and I. Wieczorek, “Resource Modeling in Software Engineering”, Marciniak J. editor. Encyclopedia of Software Engineering, 2nd edition: Wiley-Interscience: 2002, pp. 1-67.

[3] K. Moløkken and M. Jørgensen, “A Review of Software Surveys on Software Effort Estimation”, Proceedings of International Symposium on Empirical Software Engineering, Rome, Italy, 2003, pp. 223–230.

[4] M. Jørgensen, MJ. Shepperd, “A Systematic Review of Software Development Cost Estimation Studies”, IEEE Transactions on Software Engineering,Washington DC: IEEE Computer Press, vol. 33, 2007, pp. 33-53.

[5] YF. Li, M. Xie and TN. Goh, “Adaptive Ridge Regression System for Software Cost Estimating on Multi-Collinear Datasets”, Journal of Systems and Software, vol. 83, pp. 2332–2343, Novembre 2010.

[6] V. Nguyen, B. Steece, BW. Boehm, “A Constrained Regression Technique for COCOMO Calibration”. Proceedings of the 2nd ACM-IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM'08), 2008, pp. 213-222.

[7] J. Wen, S. Li, Z. Lin, Y. Hu and C. Huang, “Systematic literature review of machine learning based software development effort estimation models”, Information and Software Technology, vol. 54, doi: 10.1016/j.infsof.2011.09.002, 2012, pp. 41–59.

[8] A. Idri, F.A. Amazal, A. Abran, “Analogy-based software development effort estimation: a systematic mapping and review”, Information and Software Technology, vol. 58, doi: 10.1016/j.infsof.2014.07.013, 2014, pp. 206–230.

[9] F.A. Amazal, A. Idri and A. Abran, "Improving fuzzy analogy based software development effort estimation", In: Asian Pacific Software Engineering Conference (APSEC), Jeju, Korea, 2014, pp. 270–277 .

[10] M. Azzeh, D. Neagu and P. Cowling, "Improving Analogy Software Effort Estimation using Fuzzy Feature Subset Selection Algorithm", Proceedings of 4th International Workshop on Predictor Models in Software Engineering (PROMISE 2008), 2008, pp.71-78.

[11] Y. F. Li, M. Xie and T. N. Goh, "A study of mutual information based feature selection for case based reasoning in software cost estimation", Expert Systems with Applications, 2009, pp. 5921–5931.

[12] M. Azzeh, Menzies T, M. Shepperd, "A replicated assessment and comparison of adaptation techniques for analogy-based effort estimation, Empirical Software Engineering", 2012, pp. 90-127.

[13] C.J. Hsu and C.Y. Huang, "Comparison of weighted grey relational analysis for software effort estimation", Software Quality Journal, vol. 19, 2010, pp. 165–200.

[14] Z. Chen, T. Menzies, D. Port and B. Boehm, "Feature subset selection can improve SDEE accuracy", in: The 2005 International Conference on Predictor Models in Software Engineering (PROMISE'05), St. Louis, MO, USA, 2005, pp. 1–6.

[15] S. Das, "Filters, wrappers and a boosting-based hybrid for feature selection", in: The 18th International Conference on Machine Learning (ICML'01), Williamstown, MA, USA,2001, pp. 74–81.

[16] T.W.S. Chow and D. Huang, "Estimating optimal feature subsets using efficient estimation of high-dimensional mutual information", IEEE Trans. Neural. Netw. 16, pp, 2005, pp. 213–224.

[17] V. Sindhwani, S. Rakshit, D. Deodhare, D. Erdogmus, J. Principe and P. Niyogi, "Feature selection in MLPs and SVMs based on maximum output information'', IEEE Trans Neural Netw, vol. 15, 2004, pp: 937–948.

[18] M.D. Plumbley and E. Oja, ''A nonnegative PCA algorithm for independent component analysis'', IEEE Trans Neural Netw 15, 2004, pp: 66–76.

[19] M. Hall, "Correlation-based feature selection for machine learning", PhD Thesis., Department of Computer Science, Waikato University, New Zealand, 1990.

[20] J.H. Holland,''Adaptation in Natural and Artificial Systems''. University of Michigan Press, Ann Arbor, 1975.

[21] R. O. Duda, P. E. Hart, D. G. Stork, ''Pattern Classification''. Wiley, New York, 2001.

[22] I. Guyon and A. Elisseeff, "An introduction to variable and feature Selection", Journal of Machine Learning Research., vol. 3, 2003, pp. 1157–1182.

[23] I. Guyon et al., "Gene selection for cancer classification using support vector machines", Machine Learning, vol. 46, 2002, pp. 389–422.

[24] Huang J., Li YF. and Xie M., "An empirical analysis of data preprocessing for machine learning-based software cost estimation" , information and software technology, 2015, pp. 108-127.

[25] K. Dejaeger, W. Verbeke, D. Martens and B. Baesens, "Data mining techniques for software effort estimation: A comparative study", IEEE Transactions on Software Engineering, vol. 38, 2012, pp. 375 – 397.

[26] E. Papatheocharous, H. Papadopoulos and A.S. Andreou, "Feature subset selection for software cost modelling and estimation", Engineering Intelligent Systems, vol. 30, 2010, pp. 1-6.

[27] E. Kocaguneli, S. Member and T. Menzies, "Exploiting the essential assumptions of analogy-based effort estimation", IEEE Transactions on Software Engineering, vol. 38, 2014, pp. 425–439.

[28] C. Kirsopp, MJ. Shepperd and J. Hart, "Search Heuristics, Case-Based Reasoning and Software Project Effort Prediction", Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'02), New York, NY, USA: 2002, pp.1367-1374.

[29] O. Jalali, T. Menzies, D. Baker and J. Hihn, "Column pruning beats stratification in effort estimation", Proceedings - ICSE 2007 Workshops: Third International Workshop on Predictor Models in Software Engineering, 2007.

[30] GH. Subramanian and S.Breslawski. "Dimensionality Reduction in Software Development Effort Estimation", Journal of Systems and Software, 1993, pp. 187-196.

[31] M. Shepperd and C. Schofield, "Estimating Software Project Effort Using Analogies", IEEE Transactions on Software Engineering, vol. 23, pp. 736-743,1997.

[32] T. Menzies, D. Port, Z. Chen and J. Hihn, "Specialization and Extrapolation of Software Cost Models", Proceedings of the 20th IEEE/ACM International Conference on Automated Software Engineering, 2005, pp. 384-387.

[33] J.W. Keung, B. Kitchenham and D.R. Jeffery, "Analogy-X: providing statistical inference to analogy-based software cost estimation", IEEE Transactions on Software Engineering, vol. 34, 2008, pp. 471–484.

[34] I. Stamelos, L. Angelis, M. Morisio, E. Sakellarisc and G.L. Bleris, "Estimating the development cost of custom software", Information Management, vol. 40, 2003, pp. 729–741.

[35] A. Idri, A. Abran and S. Robert, "Investigating soft computing in case-based reasoning for software cost estimation", International Journal of Intelligent Systems, 2002, pp1–18.

[36] M.A. Sicilia, J.J. Cuadrado, J. Crespo and E. García-Barriocanal, "Software cost estimation with fuzzy inputs modelling and aggregation cost drivers", Kybernetika, vol. 41, 2005, pp. 249–264.

[37] A. Idri, A. Zahi and A. Abran, "Generating fuzzy term sets for software project attributes using fuzzy c-means and real coded genetic algorithms", Information and Communication Technology in Muslim World, 2012.

[38] D. Port and M. Korte, "Comparative studies of the model evaluation criterions MMRE and PRED in SDEE research", Proceedings of the 2nd ACM-IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM'08), 2008, pp. 51-60.

[39] J.C. Bezdek, "A convergence theorem for the fuzzy isodata clustering algorithms", IEEE Transactions on Pattern Analysis and Machine Intelligence. PAMI-2, doi: 10.1109/TPAMI, 4766964, 1980, pp. 1–8,.

[40] X.L. Xie and G. Beni, "A validity measure for fuzzy clustering", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 13, doi: 10.1109/34.85677, 1993, pp. 841–847.

[41] F. Herrera, M. Lozano and A. Sánchez, "A taxonomy for the crossover operator for real-coded genetic algorithms: an experimental study", International Journal of Intelligent Systems, vol. 18, 2003, pp. 309–338.

[42] H. Mühlenbein and D. Schlierkamp-Voosen, "Predictive models for the breeder genetic algorithm I", Continuous Parameter Optimization. Evolutionary Computation, vol. 1,1993, pp. 25–49.

[43] A. Idri and A. Abran, "A fuzzy logic based set of measures for software project similarity: validation and possible improvements, In:Proceedings. Seventh International Software Metrics Symposium" doi: 10.1109/METRIC.2001.915518, 2001, pp. 85-96.

[44] Tim Menzies, Carter Pape, Corbin Steele, Mitch Rees-Jones, David Pryor, The PROMISE repository of SE data. http://openscience.us/repo/effort/.

[45] A. Idri, A. Abran and S. Mbarki, ''An experiment on the design of radial ba- sis function neural networks for software cost estimation'', In: 2nd International Conference on Information & Communication Technologies, vol. 1, doi: 10.1109/ICTTA.2006a.1684625, 2006, pp. 1612–1617.

[46] M. Shepperd and S. MacDonell, ''Evaluating prediction systems in software project estimation'', Information and Software Technology, vol. 54, doi: 10.1016/j.infsof.2011.12.008, 2012, pp. 820–827.

[47] S.D. Conte, H.E. Dunsmore and V.Y.Shen. "Software Engineering Metrics and Models", The Benjamin/Cummings Publishing Company, Inc. 1986.