Cuadrado-Gallego JJ, Rodríguez D, Sicilia MA *et al.* Software project effort estimation based on multiple parametric models generated through data clustering. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 22(3): 371~378 May 2007

# Software Project Effort Estimation Based on Multiple Parametric Models Generated Through Data Clustering

Juan J. Cuadrado Gallego[1], Daniel Rodríguez[1], Miguel Ángel Sicilia[1], Miguel Garre Rubio[1] and Angel García Crespo[2]

[1] *Department of Computer Science, The University of Alcalá, Alcalá, Spain*

[2] *Department of Computer Science, Carlos III University, Madrid, Spain*

E-mail: {jjcg, daniel.rodriguezg, msicila, miguel.garre}@uah.es; acrespo@ia.uc3m.es

Received March 15, 2006; revised February 15, 2007.

**Abstract** Parametric software effort estimation models usually consists of only a single mathematical relationship. With the advent of software repositories containing data from heterogeneous projects, these types of models suffer from poor adjustment and predictive accuracy. One possible way to alleviate this problem is the use of a set of mathematical equations obtained through dividing of the historical project datasets according to different parameters into subdatasets called partitions. In turn, partitions are divided into clusters that serve as a tool for more accurate models. In this paper, we describe the process, tool and results of such approach through a case study using a publicly available repository, ISBSG. Results suggest the adequacy of the technique as an extension of existing single-expression models without making the estimation process much more complex that uses a single estimation model. A tool to support the process is also presented.

**Keywords** software engineering, software measurement, effort estimation, clustering

## 1 Introduction

Parametric estimation techniques are nowadays widely used to measure and/or estimate the cost associated to software development[1]. The Parametric Estimating Handbook[2] defines parametric estimation as "a technique employing one or more cost estimating relationships and associated mathematical relationships and logic". Parametric techniques are based on identifying variables that obtain numerical estimates from main input variables that are known to affect the effort or time spent in development.

One important aspect of the process of deriving models from databases is that of the heterogeneity of data. A measure of such heterogeneity is heteroscedasticity, i.e., non–uniform variance. It is well-known that heteroscedasticity is a problem affecting data sets that combine data from heterogeneous sources[3]. As a result, when using such software engineering databases, traditional application of regression equations to derive a single mathematical model results in poor adjustment to data and subsequent potential high deviations. This is due to the fact that a single model cannot capture the diversity of distribution of different segments of the database points. As an illustrative example, the straightforward application of a standard least squares regression algorithm to the points used in the reality tool of the ISBSG 8 database distribution results in measures of $MMRE = 2.8$ and $Pred(0.3) = 23\%$ (these measures are introduced later), which are poor figures of predictive quality.

The use of clustering techniques has been described as a solution to provide more realism to parametric models by decomposing the model in a number of sub–models, that are used for project estimation[4] with improved accuracy when compared with single models. The resulting predictive schemes have been called *segmented* models. One of the principal benefits of this kind of techniques is the fact that the search of segmented models satisfying some pre-established quality conditions can be automated through existing clustering methods.

The rest of this paper is structured as follows. Section 2 describes related work in segmented models for software estimation. The process for software project estimation using clustering techniques and associated tool is described in Section 3. Then, Section 4 reports on empirical work performed to validate the process using a publicly available repository. Finally, conclusions and future work are discussed in Section 5.

## 2 Related Work

Shepperd *et al.*[5] classify estimation and prediction techniques into three main categories: (i) expert judgement; (ii) algorithmic models; and (iii) machine learning. This work focuses on combining clustering as machine learning technique with classical regression models. The use of different clustering approaches has already been applied to several aspects of software management, including software estimation, software quality and software metrics.

Xu and Khoshgoftaar[6] use the fuzzy c-means algorithm for variable, the partitioning of the data into a number of clusters based on experiences.

Pedrycz and Succi[7] also use fuzzy c-means as a tool to derive prototypes related to software code measurements. Dick *et al.*[8] use the same algorithm for a simi-

Regular Paper

This work is supported by the Spanish Ministry of Science and Technology under Grant No. CICYT TIN2004-06689-C03.

repancies among the different clusters. A global evaluation and comparison with single models is discussed in the next subsection.

### 4.3 Testing Process

Once we have obtained a regression for each cluster, we need to perform an overall evaluation of the accuracy of these multiple models and its comparison against single models.

The testing process is performed in the same way as the partitioning and clustering process. The testing dataset, which is composed of 90 projects, is divided into four partitions taking into account the METHO and CASET attributes. Then, each partition will be used to test the corresponding clusters, for example, the partition with METHO=yes and CASET=yes will be used to test clusters #1 and #2.

For the single model, only one value of *MMRE* and *Pred*(%) indicates the predicted accuracy of the whole model using single cross validation. However, when multiple models are used, each regression curve has an associated *MMRE* and *Pred* values that needs to be averaged to obtain the accuracy of the whole model.

Table 3 compares the *MMRE* and *Pred* values when a single model is used to the averaged *MMRE* and *Pred* (see Table 2) when multiple models through clustering are used. As it can be observed, both *MMRE* and *Pred* have been significantly improved. The *MMRE* value indicates that the accuracy is doubled and *Pred* is improved in around 10%. Although this improvement is not good enough for software engineering effort estimation standards due to the heterogeneity of the ISBSG repository, the above commented results of this process suggests that this is an valid approach. Furthermore, both prediction values improve, this is not always the case when applying machine learning techniques to software effort estimation.

**Table 3.** *MMRE* and *Pred* Comparison of a Single Model vs. Multiple Models

| | *MMRE* | *Pred* (< 0.3) (%) |
|---|---|---|
| Single Model | 2.17 | 26.75 |
| Using Clustering | 1.03 | 35.60 |

## 5 Conclusions and Future Work

This paper presented a process to generate multiple regression models for software effort estimation using clustering. Such a process was validated using a publicly available repository, the International Software Benchmarking Standards Group (ISBSG) database, which provides software management data from multiple organizations. A tool, called RCT, was also presented to facilitate the estimation process using this technique.

The estimation process proposed here consists of the following steps: (i) the project management repository is divided into partitions according to important attributes; (ii) subsequently, clusters are obtained for each partition, in this work, using the EM algorithm; (iii) then, a regression equation is calculated for each cluster; (iv) finally, once we need to perform a new estimate, a regression equation will be selected according to the available data.

From the results and observations performed in the experimental work, it is possible to conclude that the accuracy of the estimates is improved when compared to single models. We believe that this accuracy can be greatly improved if further attributes were taken into account for both the clustering process and regression equations.

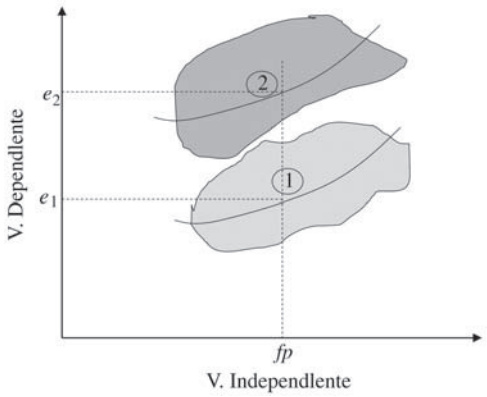


Fig.6. Superimposed clusters.

For future work, we intend to perform further empirical studies taking into account further attributes for clustering and regression. This will not only improve the accuracy of estimates, but also alleviate problems when clusters are superimposed. For example, as shown in Fig.6, when only effort ($e$) and size ($fp$) are used, there could be two possible values for the effort ($e_1$ y $e_2$). Further work will also compare this technique with other machine learning techniques such as estimation by analogy[11].

## References

[1] Boehm B, Abts C, Chulani S. Software development cost estimation approaches — A survey. USC Center for Software Engineering Technical Report USC-CSE-2000-505, 2000.

[2] Parametric Estimating Initiative. Parametric Estimating Handbook, 2nd Edition, 1999.

[3] Stensrud E, Foss T, Kitchenham B, Myrtveit I. An empirical validation of the relationship between the magnitude of relative error and project size. In *Proc. the Eighth IEEE Symp. Software Metrics*, Ottawa, Canada, 2002, pp.3–12.

[4] Cuadrado-Gallego J J, Sicilia M A, Garre M *et al.* An empirical study of process-related attributes in segmented software cost-estimation relationships. *Journal of Systems and Software*, 2006, 79(3): 351∼361.

[5] Shepperd M, Schofield C, Kitchenham B. Effort estimation using analogy. In *Proc. 8th Int. Conf. Software Engineering*, IEEE Computer Society Press, Berlin, 1996, pp.170∼178.

[6] Xu Z, Khoshgoftaar T. Identification of fuzzy models of software cost estimation. *Fuzzy Sets and Systems*, 2004, 145(1): 141∼163.

[7] Pedrycz W, Succi G. Genetic granular classifiers in modeling software quality. *The Journal of Systems and Software*, 2002, 76(3): 277∼285.
[8] Dick S, Meeks A, Last M *et al.* Data mining in software metrics databases. *Fuzzy Sets and Systems*, 2004, 145(1): 81∼110.
[9] Lung C H, Zaman M, Nandi A. Applications of clustering techniques to software partitioning, recovery and restructuring. *Journal of Systems and Software*, 2004, 73(2): 227∼244
[10] Dolado J. On the problem of the software cost function. *Information and Software Technology*, 2001, 43(1): 61∼72.
[11] Shepperd M, Schofield C. Estimating software project effort using analogies. *IEEE Trans. Software Engineering*, 1997, 23(11): 736∼743.
[12] Oligny S, Bourque P, Abran A, Fournier B. Exploring the relation between effort and duration in software engineering project. In *Proc. World Computer Congress*, Beijing, China, August 21∼25, 2000, pp.175∼178.
[13] Marquardt W. An algorithm for least squares estimation of non-linear parameters. *J. Soc. Indust. Appl. Math.*, 1963, 11: 431∼441.
[14] Conte S D, Dunsmore H E, Shen V Y. Software Engineering Metrics and Models. Menlo Park: Benjamin/Cummings, CA, 1986.
[15] Kohavi R, John G. Automatic parameter selection by minimizing estimated error. In *Proc. 12th Int. Conf. Machine Learning*, San Francisco, 1995, pp.304∼312.
[16] Witten I H, Frank E. Data Mining, Practical Machine Learning Tools and Techniques with Java Implementations. San Francisco: Morgan Kaufmann Publishers, USA, 2005.
[17] NESMA. NESMA FPA counting practices manual (CPM 2.0), 1996.
[18] Dreger J B. Function Point Analysis. Englewood Cliffs, NJ: Prentice Hall, 1989.

**Juan J. Cuadrado Gallego** is currently with the Department of Computer Science at the University of Alcalá, Madrid, Spain and the University Oberta of Catalunya, Barcelona, Spain. He previously hold positions at the University of Valladolid and Carlos III University, Madrid, Spain, where he obtained his doctorate in computer sciences engineering in 2001. His research interests are in the area of software engineering and more specifically in software measurement. He is the president of the Spanish Function Points Users Group (SFPUG).

**Daniel Rodríguez** is a lecturer in the Department of Computer Science at the University of Reading. He received his degree in computer science from the University of the Basque Country, Spain, in 1995 and his Ph.D. degree from the University of Reading, UK, in 2003. His research interests are primarily in the area of software engineering (SE) including empirical software engineering and the application of data mining to SE.

**Miguel Ángel Sicilia** obtained an M.Sc. degree in computer science from the Pontifical University of Salamanca, Madrid, Spain in 1996 and a Ph.D. degree from the Carlos III University in 2003. Currently, he leads the Information Engineering Unit at the Computer Science Department, University of Alcalá. His research interests are primarily in the areas of adaptive hypermedia, learning technology and human-computer interaction.

**Miguel Garre Rubio** received his B.Sc. degree from the University of Murcia, Spain and his doctoral degree in Computer Science from the University of Alcalá, Spain. Currently, he is with the University of Alcalá, Spain with the Open University of Spain. His research interests are in the area of software engineering and software measurement.

**Angel Garcia Crepo** is a lecturer and subdirector of teaching at Carlos III University, Madrid, Spain. He holds a doctoral degree in industrial engineering by Polytechnic University, Madrid, Spain and an MBA by the IE Business School. As a director of the advanced systems integration team, his research interest include software engineering.