Software Productivity estimation based on Association Rules

S. Bibi, I. Stamelos, L.Angelis

Abstract

Software process improvement models have as a target to help software organizations produce successfully, under the expected quality and within time and budget constraints their projects. For this purpose various steps are suggested by these models for the improvement and measurement of the processes followed. One of them involves project planning which is strongly connected with software cost or productivity estimation. Target of this study is to provide a method that could be adopted by an organization in order to estimate the required productivity for the completion of a software development project. Association Rules (A.R) is a suitable technique capable of discovering knowledge concerning productivity. The proposed method is applied and evaluated on two different data sets, namely the COCOMO81 dataset and the Maxwell dataset. The evaluation shows that A.R is a promising method whose results can be confirmed intuitively.

Keywords
Cost estimation, Intervals, Association Rules, COCOMO.

1 Introduction

Software process improvement models include a framework for planning, managing, controlling, and improving the development, operation and support of software. Their target is to identify key process areas and practices that may comprise successful and within time and budget constraints projects. Common features of these models include ability to perform and measurement and analysis, two features that involve project planning and cost estimation. Analyzing and summarizing data for cost and schedule estimation and using collected metrics to calibrate and update software estimate models are key practices that could improve a part of the software process.

An estimation of productivity, effort or duration needed to complete a project would help organize and distribute the required work. The above estimation is characterized by difficulties such as poorly defined requirements, frequent staff turnover and volatile software platforms and therefore should take into consideration uncertainy and risk.

A technique that takes into consideration the above issues is Association Rules (A.R) [3]. A.R. can be applied on past historical data in order to discover knowledge concerning productivity and deals with uncertainty and risk in two different ways. The first one is presented by two probabilities that accompany each rule. These probabilities express the validity of the rule and its frequency in the dataset, providing a measure of appropriateness of each rule. The second way refers to the estimation of productivity intervals. Productivity values are quantified into categories having as a result the estimation of productivity intervals. Intervals give a pessimistic estimate and an optimistic estimate between which the actual productivity of a project may fall in.

Purpose of this study is to extract useful patterns from cost estimation data with the help of Association Rules and to provide some evidence of the prediction accuracy of this technique. Also some conclusions will be drawn concerning the factors that tend to affect productivity directly. The method is applied on two different data sets, namely the widely known COCOMO81 dataset and the Maxwell dataset.

Various studies have been conducted so far concerning the comparison and evaluation of different cost estimation techniques [4], [6], [8], [10], [11]. In particular, some of them suggest the estimation of intervals [1], [5], [12]. Effort estimation with the help of A.R is presented in [8] where a limited number of rules are extracted from decision trees. The results of rule induction are not encouraging as rules are used as representation method, not as a modelling technique. .In our approach we propose that rules be used both as a modelling and representation method allowing the extraction of many useful patterns.

2 Modelling technique and results

2.1 Modelling technique and methodology

The modeling technique is Association Rules. A.R [3] belong to descriptive modeling and have as a target to describe the data and their underlying relationships with a set of rules that jointly define the target variables. An association rule is a simple probabilistic statement about the co-occurence of certain events in a database. A simple association rule has the following form: IF A1=X AND A2=Y THEN A3=Z with probability p (Confidence). There is also one more probability that accompanies each rule: Support. Support is a measure that expresses the frequency of the rule in the whole data set. As a consequence, the rules that are extracted from a dataset are ranged hierarchically according to their confidence at first and then by their support.

The data sets used in order to extract A.R with the help of an open source tool [7] are COCOMO81 data set and the Maxwell data set. The data sets are analytically presented in [6], [9].

Before extracting A.R, productivity that is a continuous variable, had to be quantified into discrete categories. We preferred to consider intervals that may be appealing to software managers: relatively few intervals were chosen (because of the low number of projects in the datasets) with rounded lower and upper limits, so as to be easily identifiable by a human. Productivity intervals for both data sets are presented in [1].

2.2 Results

The accuracy metrics that will be used in order to evaluate the results of each model are the Mean Magnitude Relative (MMRE), the PRED(Y) and the hitrate. $MMRE = \frac{100}{n}\sum_{i=1}^{n}\frac{|P_i - E_i|}{Pi}$ where P_i is the actual productivity and E_i is the estimate and n is the number of projects.. $PRED(Y)$ is the percentage of projects (k) for which the prediction falls within the Y% of the actual value. Relative errors are calculated by considering the mean of the interval in order to derive a point estimate from an interval estimate. Finally, *hitrate* will be used [5] i.e. the percentage of projects for which the correct interval has been successfully estimated. Usually the validation of a model is done by removing one data point at a time from the data set, recalculating the model and estimating the value of the project that was left out (a method known as JackKnifing).

While extracting A.R the following issues arose: The most representative and powerful, rules should

be selected. In addition, the selected rules should be able to provide estimates for all possible projects. In order to achieve this, the rules had to be as general as possible, with few constraints in their rule body, so as, given the attributes of a new unknown project, to be able to provide an estimate. For that purpose, rules with high confidence and as high support as possible have been preferred. In order to satisfy the second constraint, rules with few attributes in the rule head were selected so as to avoid over specialization. Eventually, for COCOMO81 data set 36 rules were selected with support threshold 4,7% (3 projects), which were used for the evaluation of the model and confidence threshold 50%. For instance, two rules concerning two categories of productivity are the following in priority order (PROD_i corresponds to productivity interval *i* , The intervals are presented in [1]):

Support	Confidence	Rule Body	Rule Head
6.3	80.0	NOM +H_VH_EH_SH_RVOL+ H_VH_TURN	==>PROD_3]
6.3	66.6	ACAP_H + DATA_N	==> PROD_4]

As an example, the second rule can be interpreted as follows: When the programmers' analysis capability is high and the database size is nominal then the productivity is likely to be in the fourth category (100<PROD<160). This pattern is presented in 6.3% of the dataset projects (4 projects) and 66.7% of the projects that present ACAP high and DATA nominal fall into the fourth category of productivity (4 out of 6 projects). The rule implies that, in the given cost database, the combination of these two conditions is sufficient to suggest a plausible productivity level, with a certain probability. The estimate is probabilistic in nature because in various cases other project factors may also rise or lower productivity. Notice that a new project with attributes that satisfy both rules will actually receive two estimate intervals from which the one is more likely to appear than the other. For both datasets the evaluation results are presented in table1.

Table 1: Results of the model for both datasets.

	COCOMO81 Jacknifing method	Maxwell data set Projects with starting dates 1992,1993 are estimated	Maxwell data set JackKnifing method
HITRATE	76.1	66.6	63.33
PRED(25) %	63.4	75	60
MMRE %	29.5	23.5	42.5

While observing carefully the extracted rules it appears that RELY, MODE, CPLX, PCAP and LEXP are the attributes that appear in the majority of the rules defining the productivity of a project. PRED(20)= 55.5 is also calculated in order to compare the results with other studies. The original Intermediate COCOMO81 model, estimating project effort, has a PRED(20) equal to 68 percent and a MMRE equal to 18.4%. However, this model is constructed in an *ad hoc* way not easily repeated in other databases. In [6], where Forward Pass Residual Analysis was performed the results indicate a MMRE of 36%,and a PRED(20) of 49% .

In the Maxwell dataset, 36 rules are extracted with support threshold 5.0% (3 rules) and confidence threshold 40%. The same criteria used in the COCOMO81 dataset are used in order to prune the rules. The are presented in table 1.

Frequently met attributes in the rules are CPLX, Installation Requirements, Efficiency Requirements, Staff Tool Skills and Staff Team Skills.

In [9], where the dataset was published, regression was applied on the projects with starting dates

before 1992 and the projects with starting dates 1992,1993 were evaluated. For this model, the dependent variable was project effort. The model is within 25 percent of the actual effort 58 percent of the time and in that case has a MMRE equal to 32%.

It should be mentioned that, when leaving out of the study the most often observed variables, the evaluation results were disappointing indicating that some times the rules suggest causal relationships.

3 Conclusions and future work

In this paper Association Rules are proposed for creating a model predicting productivity. A.R have been implemented and evaluated on two public datasets, testing the accuracy of the method and its suitability.

In both data sets the results are competitive with those obtained through conventional techniques and promising for further evaluation. Because the models proposed in [2], [6] and [9] predict effort and not productivity, only tentative comparisons may be made. In the case of the COCOMO81 dataset the prediction accuracy of A.R. is slightly lower than that of the original COCOMO81 intermediate model. However, it must be stressed that the latter is an ad hoc model, and cannot be validated through JackKnifing. Regarding the model presented in [6] A.R produce competitive results with a noticeable improvement in the estimation accuracy. In the case of the Maxwell dataset, A.R. JackKnifing accuracy is relatively low, but is acceptable for the 1992, 1993 projects and much better than that of Maxwell's model. In addition, when Hitrate is considered, which is a more appropriate and fair method to evaluate interval estimations, the results show that in the majority of cases, A.R predictions fall into the correct interval of productivity and are able to guide software managers in staffing their project and defining the time schedule of their project.

Regarding the advantages of A.R, it should be pointed out that they are one of the most expressive and human readable representations for learned hypotheses in sets of if-then rules, expressing uncertainty in many ways. First, by considering productivity intervals, and secondly, by characterizing each rule with two probability values. Also A.R's performance can be improved easily by using expert judgment as a support for pruning of the final rule set and for the initial selection of productivity intervals.

An issue that deserves further attention is the fact that the pruned A.R cannot cover all the cases that may appear. In addition, A.R present the usual drawback of all machine learning techniques, i.e. the possibility of over specialization of the training data. Future research also needs to focus on confirming and enriching the results of A.R in larger, multi-organizational datasets, such as those coming from ISBSG .

4 Literature

1. Bibi, S., Stamelos, I. Angelis, L.: Software Cost Prediction with Predefined Interval Estimates, 1st Software Measurement European Forum, Rome, Italy, January 2004.

2. Boehm, B.: Software Engineering Economics, Prentice-Hall, NJ, (1981)

3. Hand, D., Mannila, H., Smyth, P.: Principles of Data Mining, MIT Press, US, (2001)

4. Jeffery, R., Ruhe, M., Wieczorek, I.: Using Public Domain Metrics to Estimate Software Development Effort, in IEEE 7th International Software Metrics Symposium proceedings, London UK (2001) 16-27

5. Jorgensen, M.: An effort prediction interval approach based on the empirical distribution of previous estimation accuracy, Information and Software Technology 45, (2003) 123-126

6. Kitchenham, B.: A procedure for analyzing unbalanced datasets, IEEE Transactions on Software Engineering 24 (4), 1998 278-301

7. Machine learning software in Java, http://www.cs.waikato.ac.nz/ml/weka/.

8. Mair, C., Kadoda, G., Lefley, M., Phalp, K., Schofield, C., Shepperd, M., Webster, S., "An investigation of machine learning based prediction systems", Journal of Systems and Software 53, (2000) 23–29

9. Maxwell, K. Applied Statistics for Software Managers, Prentice-Hall, N. J, (2002)

10. Briand L., Emam, K., Surmann, D., Wieczorek, I. "An Assessment and Comparison of Common Software Cost Estimation Modeling Techniques", in 21st International Conference on Software Engineering proceedings , Los Angeles, California, United States (1999) 313-322

11. Srinivisan, K., Fisher, D., "Machine learning approaches to estimating software development effort", IEEE Transactions on Software Engineering 21(2), 1995, pp. 126-137.

12. Stamelos, I., Angelis, L., Dimou, P., Sakellaris, E., "On the use of Bayesian belief networks for the prediction of software productivity", Information and Software Technology 45, (2003) 51-60.

5 Author CVs

Stamatia Bibi

Bibi Stamatia received her BsC from the department of Informatics at the Aristotle University of Thessaloniki, Greece. Currently, she is a PhD student at the Software Engineering Laboratory of the same department. Her research interests involve software metrics and software process improvement. In particular she is comparing estimation methods and is pursuing novel estimation techniques for software cost prediction. In that direction, she has published a study on the application of Bayesian Belief Networks for cost estimation and another one on the estimation of the development cost for an open source ERP system.

Ioannis Stamelos

Ioannis G. Stamelos is an Assistant Professor of computer science at the Aristotle University of Thessaloniki, Dept. of Informatics. He received a degree in Electrical Engineering from the Polytechnic School of Thessaloniki (1983) and the Ph. D. degree in computer science from the Aristotle University of Thessaloniki (1988). He teaches language theory, object-oriented programming, software engineering, software project management and enterprise information systems at the graduate and postgraduate level. His research interests include empirical software evaluation and management, software education and open source software engineering. He is author of 50 scientific papers and member of the IEEE Computer Society.

Lefteris Angelis

Lefteris Angelis received his BSc and Ph.D. degree in Mathematics from Aristotle University of Thessaloniki (A.U.Th.). He works currently as an Assistant Professor at the Department of Informatics of A.U.Th.. His research interests involve statistical methods with applications in software engineering and information systems, computational methods in mathematics and statistics, planning of experiments and simulation techniques.