Information and Software Technology xxx (2014) xxx–xxx

Contents lists available at ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof

# A framework for comparing multiple cost estimation methods using an automated visualization toolkit

Nikolaos Mittas [a], Ioannis Mamalikidis [b], Lefteris Angelis [a,*]

[a] Department of Informatics, Aristotle University of Thessaloniki, 54124 Thessaloniki, Greece
[b] Department of Electrical Engineering, Technological Educational Institute of Kavala, 65404 Kavala, Greece

A R T I C L E I N F O

*Article history:*
Received 27 September 2013
Received in revised form 12 May 2014
Accepted 15 May 2014
Available online xxxx

*Keywords:*
Software Cost Estimation
Prediction models
Graphical comparison
REC curves
Automated tools

A B S T R A C T

*Context:* The importance of accurate predictions in Software Cost Estimation and the related challenging research problems, led to the introduction of a plethora of methodologies in literature. However, the wide variety of cost estimation methods, the techniques for improving them and the different measures of accuracy have caused new problems such as the inconsistent findings and the conclusion instability. Today, there is a confusion regarding the choice of the most appropriate method for a specific dataset and therefore a need for well-established statistical frameworks as well as for automated tools that will reinforce and lead a comprehensive experimentation and comparison process, based on the thorough study of the cost estimation errors.
*Objective:* The purpose of this paper is to present a framework for visualization and statistical comparison of the errors of several cost estimation methods. It is based on an automated tool which can facilitate strategies for an intelligent decision-making.
*Method:* A systematic procedure comprised of a series of steps corresponding to research questions is proposed. For each of the steps, StatREC, a Graphical User Interface statistical toolkit is utilized. StatREC was designed and developed to take as input a simple data matrix of predictions by multiple models and to provide a variety of graphical tools and statistical hypothesis tests for aiding the users to answer the questions and choose the appropriate model themselves.
*Results:* The study of prediction errors by the proposed framework provides insight of several aspects related to prediction performance of different models. The systematic examination of candidate models by a series of research questions supports the user to make the final decision.
*Conclusion:* Structured procedures based on automated tools like StatREC can efficiently be used for studying the error and comparing cost estimation models.



## 1. Introduction

The estimation of cost for software projects has been an open research problem for decades, due to its vital role in all decisions concerning the software development management. The complexity and the interrelations of software development procedures, which are different even within the same organization, make the accurate prediction of cost a challenging problem. A software project is often more expensive than estimated and it is completed later than planned [14], while the consequences of underestimations, or even overestimations, and the associated delays can be catastrophic for both developers and customers.

Due to these requirements and needs, there has been a growing interest, reflected in the related literature [18], regarding the identification of the most “accurate” or “best” *Software Cost Estimation*

*Abbreviations:* ANOVA, Analysis of Variance; AOC, Area Over Curve; AUC, Area Under Curve; BRACE, Bootstrap based Analogy Cost Estimation; CIs, Confidence Intervals; COCOMO, Constructive Cost Model; CDF, Cumulative Distribution Function; DOE, Design of Experiments; SLOC, equivalent physical 1000 lines of source code; ESS, Explained Sum of Squares; GUI, Graphical User Interface; ISBSG, International Software Benchmarking Standards Group; KS, Kolmogorov-Smirnov; LOOCV, Leave-One-Out-Cross-Validation; MMRE, Mean Magnitude of Relative Error; MMER, Mean Magnitude of Relative Error to the Estimate; PRICE-S, Programming Review of Information Costing and Evaluation-Software; RCBD, Randomized Complete Block Design; ROC, Receiver Operating Characteristic; REC, Regression Error Characteristic; RSS, Residual Sum of Squares; SK, Scott–Knott; SCE, Software Cost Estimation; SLIM, Software Life-cycle Model; SPR, Software Productivity Research; StatREC, Statistical Regression Error Characteristic; STAINS, Statistics and Information Systems; TSS, Total Sum of Squares.

* Corresponding author. Tel./fax: +30 2310998230.
*E-mail address:* lef@csd.auth.gr (L. Angelis).

http://dx.doi.org/10.1016/j.infsof.2014.05.010
0950-5849/© 2014 Elsevier B.V. All rights reserved.

(SCE) technique. The proposed methods ranging from *expert judgment* to *machine learning* can certainly provide valuable aid in decision-making and management of the whole development process. However, the plethora and diversity of SCE models are causes of confusion to the inferential mechanisms since there are no clear instructions and recommendations about the strategy that a decision maker should follow. Indeed, the findings are frequently contradictory concerning the superiority of a technique against a comparative one and there is no global answer to the critical issue of the identification of the "best" prediction technique. During the past few years, the problem of conclusion instability opened new directions in the SCE research towards the analysis of the sources generating divergent results.

The discussion about the sources of conclusion instability concerns different aspects, such as: (a) the dependency of the prediction methodologies on the available data (i.e. types and number of project attributes) [44]; (b) the appropriateness of the local measures of error used [20,11]; (c) the strategy of promoting a certain prediction model against a competitive one through the usage of groundless statistical procedures [26,21] and (d) the experimental procedures [24], such as the splitting of samples into training and tests sets and the validation schema followed in order to evaluate the predictive power of a proposed model.

Despite the interesting conclusions derived from the above-mentioned research questions and the various guidelines that a practitioner should follow, the phenomenon of conclusion instability seems to remain unresolved. We believe that one of the causes of the problem is the fact that although a large body of theoretical, well-defined methodologies can be easily applied by researchers, these techniques are not practical for decision makers, since there are no available automated software tools. Without doubt, decision makers are willing to obtain significant knowledge through well-established methodologies but at the same time, they also require automated and easy-to-use tools in order to perform high-level analyses.

In fact, there are a few automated tools that can significantly reduce the required time and costs of data gathering, risk management, scenario analysis and project management; however most of them are proprietary due to the huge effort to consolidate historical projects [6]. Furthermore, these products are mainly tools for cost estimation and not for performing comparisons between different methods.

Towards the adoption of free automated tools in SCE and the necessity for systematic utilization of graphical analysis, we developed the StatREC (*Statistical Regression Error Characteristic*) software toolbox. StatREC is a *Graphical User Interface* (GUI), based on the statistical language R and implementing in a unified manner several statistical procedures able to provide a framework for efficient model comparison.

StatREC was designed in such a way that the user only needs to enter as input a file containing the predictions of multiple models without any need for knowing algorithmic or mathematical details of how these models were built and trained. The tool then is able to perform statistical and graphical analysis on the error metrics obtained by simple calculations between the predicted and the actual values. It is therefore clear that StatREC is not a tool for modeling. The predictions can be obtained by any procedure, mathematical, algorithmic or even by empirical expert opinions.

Then, StatREC provides through an interface, a set of features that can be used for comparison of two or more prediction models. Beyond the basic facilities for automated evaluation of well-known error functions and comparison with a reference model, there are advanced facilities allowing identification of factors (independent categorical variables) affecting the error, exploration of how errors vary within a certain range of the cost variable and identification of whether a prediction model is prone to over or underestimation. Additionally to the graphical representations of the aforementioned features through curves and geometrical concepts, StatREC embraces well-known traditional statistical procedures as well as modern simulation resampling techniques, such as bootstrap for the estimation of unknown parameters of error distribution (i.e. standard error, bias and confidence intervals) and permutation tests that can be used in order to carry out formal hypothesis testing.

The purpose of this paper is to present a framework in the form of structured procedure comprised of a series of research questions. These research questions start from general aspects (ranking and clustering of all the models) and continue to more focused issues (like proneness to overestimation or underestimation). In all steps of this procedure, StatREC plays a central role, in the sense that it provides the user information about each specific question. It has to be emphasized that the entire procedure is not a problem solver. It is rather an aid for making decisions which can be different by different people.

The rest of the paper is organized as follows: In Section 2, we summarize related work and contribution of our paper. In Section 3, we present in steps our proposed framework. In Section 4, we analytically describe the theoretical and statistical concepts as well as a brief description of the StatREC tool. In Section 5, we present an illustrative example of applying the framework to a historical dataset, comparing a large number of candidate models. Finally, in Section 6, we conclude by providing some useful directions for future work.

## 2. Related work and contribution

The recent technological growth and the need of even more complex systems have settled SCE, as one of the most critical phases in planning, scheduling and risk management of software projects. Despite the introduction of many prediction methodologies, it is difficult to find easy-to-use and non-proprietary software for conducting analyses closely related to the intelligent project management.

Reviewing briefly various SCE tools appeared so far in the literature, it should be mentioned first [1] COCOMO (Constructive Cost Model) which was retuned and calibrated later by Boehm et al. [3] as COCOMO II. Both tools and other both free and commercial variants (i.e. COSTAR and Cost Xpert) were developed and continually enhanced by the USC center (http://csse.usc.edu/csse) founded by B. Boehm and affiliate organizations. ANGEL [46] is a non-proprietary tool based on estimation by analogy methodology, a form of non-parametric regression. BRACE (*Bootstrap based Analogy Cost Estimation*) [49] is also a free analogy-based tool that supports the practical application of the analogy based technique and a resampling methodology, the non-parametric bootstrap [8] for calibration and evaluation of the model's accuracy.

Other well-known, however proprietary, SCE tools are:

- SLIM (*Software Life-cycle Model*) [37], developed by Quantitative Software Management (Quantitative Software Management, Inc.) [38] designed to estimate effort, schedule and defect rate.
- SPR (*Software Productivity Research*) KnowledgePlan is a knowledge-based estimation tool proposed by Jones [17] providing mechanisms in order to size projects and then estimate the effort, resources, schedule and defects at four levels of granularity (project, phase, activity and task).
- PRICE-S (*Programming Review of Information Costing and Evaluation-Software*) [36], a parametric cost model, used for estimating US DoD, NASA and other government software projects [2].
- *SEER for Software* is another tool for project planning, cost management and tracking throughout the software development life-cycle based on the original Jensen model [16] developed by Galorath (Galorath, Inc.) [12].