ELSEVIER

Information and Software Technology 42 (2000) 1009–1016

INFORMATION AND SOFTWARE TECHNOLOGY

www.elsevier.nl/locate/infsof

# A comparative study of two software development cost modeling techniques using multi-organizational and company-specific data

R. Jeffery[a,*], M. Ruhe[b], I. Wieczorek[c]

[a]Centre for Advanced Empirical, Software Research (CAESAR), University of New South Wales, Sydney, Australia
[b]Department of Computer Science, University of Kaiserslautern, 67663 Kaiserslautern, Germany
[c]Fraunhofer Institute for Experimental Software Engineering (IESE), 67661 Kaiserslautern, Germany

**Abstract**

This research examined the use of the International Software Benchmarking Standards Group (ISBSG) repository for estimating effort for software projects in an organization not involved in ISBSG. The study investigates two questions: (1) What are the differences in accuracy between ordinary least-squares (OLS) regression and Analogy-based estimation? (2) Is there a difference in accuracy between estimates derived from the multi-company ISBSG data and estimates derived from company-specific data? Regarding the first question, we found that OLS regression performed as well as Analogy-based estimation when using company-specific data for model building. Using multi-company data the OLS regression model provided significantly more accurate results than Analogy-based predictions. Addressing the second question, we found in general that models based on the company-specific data resulted in significantly more accurate estimates. © 2000 Elsevier Science B.V. All rights reserved.

*Keywords*: Software cost estimation; Cost modeling techniques; Accuracy comparison; Analogy-based estimation; Ordinary least-squares regression

## 1. Introduction

Delivering a software product on time, within budget, and to an agreed level of quality is a critical concern for many software organizations. Underestimating software costs can have detrimental effects on software quality and thus on a company's business reputation. On the other hand, overestimation of software cost can result in missed opportunities to fund other projects. In response to industry demand, many estimation techniques have been proposed during the last three decades. In order to assess the suitability of a cost modeling technique, its performance and relative merits must be compared. Normally, homogenous company-specific data are believed to form a better basis for more accurate estimates. However, those data sets are typically small and cost driver data are tailored and specific such that comparison with other organizations or across the industry is difficult. Moreover, data collection is an expensive and time-consuming process for individual organizations. Industry representative parties have addressed the problem of software data collection in the past few years with the advent of multi-organizational data sets. The collaboration of organizations (such as the International Software Benchmarking Standards Group, ISBSG) to form multi-organizational data sets provides a possibility for reduced data collection costs, faster data accumulation and shared information benefits. Therefore, the pertinent question is whether multi-organizational data is valuable for estimation. Previous studies [1,2] have shown this to be the case for organizations participating in these data repositories. A question addressed here is whether this is also the case if the organization has not participated in such defined data collection processes.

In this study, we used public domain data from the ISBSG. We compared the estimates derived from those data with estimates derived from company-specific data from an Australian company (Megatec). At the time of our analysis, Megatec did not contribute data to the ISBSG repository.

The two modeling techniques selected were: (1) OLS regression, as it is one of the most commonly applied techniques, and (2) Analogy-based estimation, whose popularity has increased in the 90s [5,16,18]. We applied different variants of the Analogical and Algorithmic Cost Estimator (ACE) algorithm to our data sets. This algorithm calculates the difference between the target project and each completed project in a database for a set of search metrics. ACE ranks the completed projects in a database according

[*] Corresponding author.
*E-mail addresses:* r.jeffery@unsw.edu.au (R. Jeffery), m_ruhe@informatik.uni-kl.de (M. Ruhe), wieczo@iese.fhg.de (I. Wieczorek).

0950-5849/00/$ - see front matter © 2000 Elsevier Science B.V. All rights reserved.
PII: S0950-5849(00)00153-1

to their similarity. The effort of the most similar project(s) is used to predict the effort for the target project. In addition, size adjustments are applied to address differences between projects.

This study is motivated by the challenge of assessing the feasibility of using multi-organization data to build cost models for organizations and the benefits gained from company-specific data collection. The study looks at the prediction accuracy of two different estimation techniques and examines their performance based on both multi-organizational and company-specific data sets. Thus, two important questions are addressed: (1) what are the differences in estimation accuracy between a traditional technique such as ordinary least-squares (OLS) regression and Analogy-based estimation? (2) Is there a difference between estimates derived from multi-company data and estimates derived from company-specific data?

This research uses an organizational data set which is not part of the multi-organization set, and therefore provides a possibly more stringent test to the use of these types of data sets than has been carried out in the past [1,2]. Furthermore, there is a difference in the quality of data collection. When collecting the Megatec project data researchers were involved in the first place and carried out extensive prior analysis [8]. Therefore, more detailed knowledge of the data context, relationships and accuracy for Megatec was present than what could be expected for any public data set. On the other hand more characteristics are measured in the public data set.

This paper starts with a discussion of related work in Section 2 followed by the presentation of the research method in Section 3 that includes a description of the data sets, the data preparation, and the estimation techniques. The results of the analysis are described in Section 4. Section 5 presents the conclusions and discussion of practical implications.

## 2. Related work

There have been two previous studies that utilized the ISBSG data set. The first one was a descriptive study done by the ISBSG itself [17]. Examples of the areas analyzed in this report are system size, project effort, and other descriptive metrics, e.g. their range, distribution, and relationships. In the second study Lokan [9] investigated the relationship between the five elements in function point analysis. This is the first application of this data set to the issue of cost estimation.

Two other pieces of research have been completed undertaking a wide-scale comparison of software cost modeling techniques. In this research two questions were investigated. What modeling techniques are likely to yield more accurate results when using typical software development cost data? What are the benefits and drawbacks of using organization-specific data as compared to a multi-organization data set? The first study [1] was based on the so-called “Laturi-database”, which included 206 business software projects from 26 companies in Finland. The second study [2] was a replication of the first study using the European Space Agency (ESA) data set [2]. At the time of the analysis the ESA data set included 166 projects mainly from the space and military domain. The projects originated from 69 different organizations coming from 10 different European countries. In both cases the research questions were addressed using organizations that contributed to the data sets. Consistently, both studies found no significant advantages using local, company-specific data to build estimates over using external, multi-organizational databases. Moreover, in general Analogy-based techniques performed significantly worse than other traditional techniques such as OLS regression and stepwise Analysis of Variance.

Another study that was published recently investigated the difference between multi-company data and estimates derived from company specific data on the ESA database [10]. In contrast to the study by Briand et al. [2], this research found better results using company specific data using stepwise Analysis of Variance. However, both analyses are differently designed which makes it very difficult to further investigate reasons for the differences in the results.

## 3. Research method

In this section, we provide descriptive statistics for the data sets used in our analysis, summarize the data preparation activities, explain the approach followed in model building and application, and introduce the reader to the estimation techniques and evaluation criteria applied.

### 3.1. The data sets

Two data sets are used in this work: ISBSG, a publicly available multi-organizational data set consisting of a total of 451 projects at the time of this study, and Megatec, a single-company data set consisting of 19 projects.

Megatec is an Australian software development organization with about 50 employees at the time of data collection (1990–1993) that developed and distributed a range of software products in Australia and the USA. It was the first software company in Australia to gain Australian Standard 3563 (IEEE-Std.-1298), a company that was highly motivated to provide good quality data and that was also interested in research results [8]. The Megatec projects' main applications are in business areas, such as financial, banking, or inventory. A full description of the data set can be obtained from Ref. [8].

The ISBSG repository (release 5 March 1998) consisted of projects from fourteen countries; Australia is the largest contributor. There are 38 metrics collected that describe each project. Unfortunately, for many of these metrics the data is not complete. Therefore, we concentrated on a subset of the

Table 1
Ratio-scaled project metrics used for the analysis (from Megatec and ISBSG data set)

| | Megatec | | | | ISBSG | | | |
|---|---|---|---|---|---|---|---|---|
| Metric | Mean | S.D. | Min | Max | Mean | S.D. | Min | Max |
| Effort (person hours) | 1947 | 3115 | 194 | 13 905 | 5201 | 8773 | 10 | 59 809 |
| System size (unadjusted function points) | 506 | 818 | 39 | 3290 | 761 | 1215 | 11 | 9803 |
| Max team size | 4 | 2 | 1 | 10 | 6 | 7 | 1 | 55 |
| PDR | 4.97 | 2.79 | 1.50 | 11.44 | 8.19 | 6.54 | 0.19 | 35.87 |

data (see Table 1, also Section 3.2). Software practitioners voluntarily submitted the projects in the ISBSG data set. The ISBSG data was collected between 1990 and 1998 using questionnaires. Each project submitted to the ISBSG repository is validated against specific quality criteria. Function points were counted largely using the IFPUG standard. There is a wide range of programming languages; the systems are mainly written using ACCESS, COBOL, NATURAL, PL/1, and TELON. The business area types are mainly management information systems, such as office information systems or transaction and production systems. Unfortunately, there is neither any data about the experience of the software developers, nor any metric that identifies the company or gives information about the organization type of the company in the repository. Therefore, research as done by Briand et al. [1], which compared estimations for multi-organization and company-specific data using only data within the repository, cannot be repeated with the ISBSG data.

A project variable that identifies different business sectors was available within the ISBSG repository but was not available for the Megatec data. Therefore, we could not consider “business sector” as a potential explanatory variable in our study even though previous work found that the business sector explains a large amount of variation in productivity [1,11].

When selecting variables for our analysis we followed the suggestion of the ISBSG group: system size, development type, language type, and development platform are recommended as important criteria for selecting projects. Other important criteria suggested are business area type, application type, and development techniques. As we are using also data from a non-contributing company the metrics of the ISBSG repository need to have counterparts in the Megatec data set. Furthermore, we added variables that are commonly used in cost estimation (e.g. team size). Finally, we chose the variables presented in Table 1 and Table 2. None of the distributions for effort, team size, and system size was normal. As can be seen in Table 1, the range for system size and effort in ISBSG is relatively wide compared to the Megatec data. Team size also shows a large difference compared with a range of 1–10 people for Megatec projects.

The project delivery rate (PDR) is used as a measure of productivity [17]. It is defined as working hours per Function Point. A high number indicates that many hours per Function Point were needed and, therefore, shows a low productivity and vice versa. The relationship between team size and PDR as well as effort is not linear. Megatec projects are generally more productive than ISBSG project (4.97 vs. 8.18).

For ISBSG significant differences in PDR were found among the development platforms PC and mainframe and also among the language types 3GL and 4GL as well as among Application Generator (ApG) and 4GL (see

Table 2
Nominal-scaled project metrics used for the analysis and mean effort/PDR for each group (missing data excluded)

| | Megatec | | | ISBSG | | |
|---|---|---|---|---|---|---|
| Metric | Value | Mean effort | Mean PDR | Values | Mean effort | Mean PDR |
| Client/server | Yes | 1319 | 8.16 | Yes | 3443 | 4.00 |
| | No | 2237 | 3.50 | No | 1967 | 5.10 |
| Language type | ApG | – | – | ApG | 9555 | 10.14 |
| | 2GL | – | – | 2GL | 3504 | 3.46 |
| | 3GL | 1841 | 5.65 | 3GL | 5273 | 10.56 |
| | 4GL | 2245 | 3.07 | 4GL | 4138 | 6.08 |
| Development platform | PC | 1319 | 8.16 | PC | 2047 | 4.01 |
| | Midrange | 2237 | 3.50 | Midrange | 5166 | 6.24 |
| | Mainframe | – | – | Mainframe | 6403 | 10.43 |

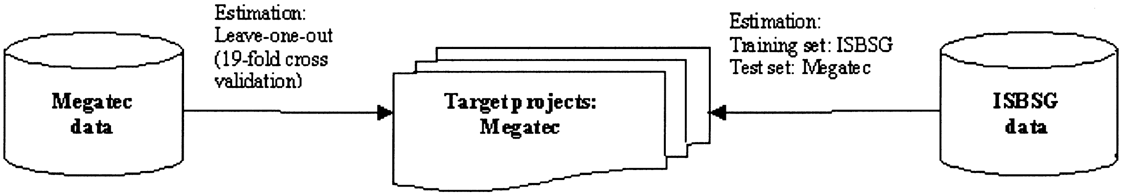

Fig. 1. Model building and application.

Table 2). There were no significant differences in effort between the different groups that are encompassed in each metric. For Megatec this kind of analysis was not applicable, as there are only 19 data points in the database.

### 3.2. Data preparation

The Megatec data is used to evaluate estimates made using the ISBSG data and to compare these with estimates made using Megatec's own data. For this study not all of the 451 ISBSG projects could be used because of the need to match characteristics of the two data sets as closely as possible. The subset used includes only those ISBSG projects that fulfill the following three criteria: Firstly, resources need to be measured on the same basis as in Megatec. Thus, ISBSG projects were included where the effort measure reflected the development team and possibly also the people supporting that development team were included. If the effort measure included operations and end user time we excluded these projects, as this effort was not included in the Megatec data set. Secondly, projects needed to have entries for the metrics system size and team size. These two variables were essential for analysis. These two qualifications resulted in a subset of 225 projects. In order to further match the characteristics of ISBSG compared with Megatec the development type was also used. The criteria development type is also one of those recommended by the ISBSG Group. Megatec projects were completely new developments whereas ISBSG projects can be new developments, redevelopments or enhancements. Furthermore, significant differences in effort as well as PDR were found between these development types. Therefore, "new development" was added as a third selection criterion. We ended up with a subset of 145 ISBSG projects.

### 3.3. Model building and application

In order to determine the accuracy of estimates based on company-specific data, we followed the 19-fold cross-validation [19]. For each of the 19 Megatec projects, we used the remaining 18 projects as a basis for model building. The overall accuracy was aggregated across the 19 projects. Calculating the accuracy in this manner emulates the situation when a company derives a cost estimation model using its own data. In order to determine the accuracy of estimates based on multi-organizational data, we used the 145 ISBSG projects as a basis for predicting the 19 Megatec projects. Thus, the cost of each target project was also estimated using the ISBSG data. This is the situation when a company uses external data to build prediction models for its own, internal projects. Fig. 1 illustrates these steps:

### 3.4. Estimation techniques applied

#### 3.4.1. Ordinary least-squares regression

OLS regression has been the most common modeling technique applied to software cost estimation [6]. It linearly approximates the relationship between one dependent variable (e.g. effort) and one or more independent variables (i.e. cost drivers) [15]. The least-squares regression method fits the data to the specified model trying to minimize the overall sum of squared errors. This is different from other techniques such as machine algorithms techniques where no model needs to be specified beforehand. In general, a linear regression equation has the following form:

$$\text{DepVar} = a + (b_1 \times \text{IndepVar}_1) + \cdots + (b_n \times \text{IndepVar}_n)$$

Where $a, b_1, ..., b_n$ are unknown parameters, DepVar stands for dependent variable and the IndepVar's are the independent variables. If the relationship is exponential, the natural logarithm is applied to the variables involved and then a linear regression equation can still be used.

There are several commonly used concepts when applying regression analysis, such as (1) the $p$-value, which indicates the probability of error of accepting the results as valid, (2) or the coefficient of determination ($R^2$), which is used to assess the percentage of variance explained by the regression model. For further details we refer the reader to Ref. [15].

The application of OLS regression makes several assumptions. For example: (1) Independent variables are not interrelated. (2) The variation in error (actual minus predicted value) is on average constant. This is called the homoscedasticity assumption. (3) Regression can only deal with interval or ratio variables, although techniques exist to include nominal or ordinal variables [7]. (4) An appropriate functional form is specified [4].

Also, regression models are sensitive to outlying observations in the training data set. This may cause misleading prediction equations not properly reflecting the trends in the data. To alleviate this problem, it is useful to identify and possibly remove outliers from the data before building an OLS regression model.

In order to comply with the assumptions and constraints stated above, a preliminary investigation on the ISBSG and Megatec data sets was performed. We performed a logarithmic transformation of the considered variables, because the data are not normally distributed. Moreover, the homoscedasticity assumption was violated when using linear model specifications. We applied a mixed stepwise regression procedure (probability to enter/leave the model: 0.05) to select variables that have a significant impact on effort (Tables 1 and 2). Nominal-scaled variables (Table 2) were coded as dummy variables [7].

### 3.5. Analogical and algorithmic cost estimator

The potential use of Analogy-based estimation for software effort estimation has been evaluated and confirmed in many studies [5,12,16,18]. Analogy is a common problem solving technique [4]. It solves a new problem by adapting solutions that were used to solve an old problem. In software effort estimation, Analogy-based estimation involves the comparison of a new (target) project with completed (source) projects. The basic idea is to identify source projects that are most similar to the new project. Major issues are (1) to select relevant project attributes (in our case cost-drivers), (2) to define an appropriate similarity function, and (3) to decide upon the number of similar source projects to consider for estimation (analogues).

Relevant project attributes may be determined by selecting the optimal combination of variables implementing a comprehensive search (ANGEL tool) [16]. This is, however, inadequate for a high number of variables and projects, as reported in Refs. [1,16]. Another strategy is proposed by Finnie et al. [5]. They applied to all categorical variables a two-tailed *t*-test to determine variables that show significant influence on productivity.

Similarity functions may be defined with the help of experts. An example of a simple measure is the unweighted Euclidean distance using variables normalized between 0 and 1 [16].

For effort prediction, one may consider one or more source projects. This decision is to be made on a case-by-case basis since no heuristic currently exists. However, studies report no significant differences in accuracy when using different numbers of analogues [1,16].

For the current study, we used a prototype tool of the Analogical and Algorithmic Cost Estimator (ACE) [18]. We considered as relevant the set of variables described in Tables 1 and 2. The similarity function is modeled through a ranking algorithm. ACE ranks all source projects in a database base according to their difference to the target project. For each considered variable, the absolute difference between the target and the source projects is calculated and the source project with the lowest difference is ranked 1 on that variable; the project with the next lowest difference is ranked 2, and so on. Then, the average rank for each source project over all variables is determined. The source project with the lowest overall rank is the most similar project. Calculating the average rank standardizes the contribution of each variable to the final ranking.

The predicted effort is determined by using the project(s) with the best rank(s). ACE adjusts the predicted effort value in order to address the differences in size between target (estimated project) and source project(s). The size adjustment is defined as:

$$\text{Effort}_{\text{ESTIMATED}}=\frac{\text{Effort}_{\text{SOURCE}}}{\text{FP}_{\text{SOURCE}}}\times\text{FP}_{\text{ESTIMATED}}$$

We applied four alternative versions of ACE in order to investigate differences in estimation accuracy when varying the number of analogues considered for prediction and the application of a size adjustment:

(1) ACE-1 no SA: considers the most similar analogue for effort prediction and does not apply any size adjustment. (2) ACE-2 no SA: uses the average of the two most similar analogues for effort prediction and does not apply any size adjustment. (3) ACE-1 with SA: uses the most similar analogue for effort prediction and applies size adjustment (see formula above). (4) ACE-2 with SA: uses the two most similar analogues for effort prediction and applies size adjustment.

### 3.6. Evaluation criteria

The evaluation of the cost estimation models was done by using the following common criteria [3]. The magnitude of relative error as a percentage of the actual effort for a project, is defined as:

$$\text{MRE}=\left|\frac{\text{Effort}_{\text{ACTUAL}}-\text{Effort}_{\text{ESTIMATED}}}{\text{Effort}_{\text{ACTUAL}}}\right|$$

The MRE is calculated for each project in the data sets. Either the mean MRE or the median MRE aggregates the multiple observations. The median MRE is less sensitive to extreme values. A mean MRE of 0.50 means that on average the estimates are within 50% of the actual values. In addition, we used the measure prediction level Pred. This measure is often used in the literature and is a proportion of a given level of accuracy:

$$\text{Pred}(l)=\frac{k}{N}$$

Where, $N$ is the total number of observations, and $k$ the number of observations with an MRE less than or equal to $l$. A common value for $l$ is 0.25, which is used for this study as well. The Pred(0.25) gives the percentage of projects that were predicted with an MRE equal or less than 0.25. Conte et al. [3] suggest an acceptable threshold value for the mean MRE to be less than 0.25 and for Pred(0.25) greater or than 0.75. In general, the accuracy of an estimation technique is proportional to the Pred(0.25) and inversely proportional to the MRE and the mean MRE.

For testing the statistical significance between paired

Table 3
Estimates based on Megatec projects applied to Megatec target projects

| Estimation method | Mean MRE | Median MRE | Pred(0.25) |
|---|---|---|---|
| ACE-1 no SA | 0.63 | 0.35 | 0.42 |
| ACE-2 no SA | 0.54 | 0.43 | 0.16 |
| ACE-1 with SA | 0.38 | 0.27 | 0.32 |
| ACE-2 with SA | 0.37 | 0.28 | 0.47 |
| OLS | 0.37 | 0.27 | 0.47 |

samples we used the two-tailed Wilcoxon signed rank test, a non-parametric analogue to the *t*-test [14].

## 4. Analysis and results

Sections 4.1 and 4.2 briefly present the results of applying the two estimation techniques on the two data sets. Section 4.3 then compares the results and, thus, builds the basis for discussion of the questions stated in Section 1.

### 4.1. Results based on Megatec data

To identify the significant variables, we performed a stepwise regression using the whole Megatec data set. Following the cross-validation approach described in Section 3.3, the identified independent variables system size and development platform were used to built a regression model for each Megatec project using the rest of the data set as a holdout sample. Applying ACE, we retrieved for each target project the most similar analogue(s) using the remaining projects as source projects.

Table 3 summarizes the aggregated results of applying OLS regression and ACE. The first column gives the estimation technique described in Section 3.4. For each technique, we provide the mean MRE, the median MRE, as well as the Pred(0.25) values. The obtained $R^2$-value for the regression model was 0.76 on average.

Looking at the median MRE values, differences in accuracy exist among the techniques but none of the differences are significant as shown and discussed later in Section 4.3. The largest difference in mean MRE observed is 0.26 (ACE-1 no SA vs. ACE-2 with SA or OLS). The differences in median MRE are lower, which is explained by outlying predictions for the Megatec data set. A detailed discussion about those outliers can be found in Ref. [13]. In general, OLS regression and ACE estimates using linear size adjustment perform slightly (though not significantly) more accurate than ACE estimates without applying size adjustment.

Table 4
Estimates based on 145 ISBSG projects applied to Megatec target projects

| Estimation method | Mean MRE | Median MRE | Pred(0.25) |
|---|---|---|---|
| ACE-1 no SA | 2.48 | 0.90 | 0.05 |
| ACE-2 no SA | 1.47 | 0.66 | 0.16 |
| ACE-1 with SA | 2.39 | 0.84 | 0.16 |
| ACE-2 with SA | 1.43 | 0.72 | 0.05 |
| OLS | 0.61 | 0.38 | 0.21 |

### 4.2. Results based on ISBSG data

Having performed a stepwise regression for the ISBSG projects the following OLS regression model was derived based on 145 ISBSG projects to predict the 19 Megatec projects:

$$\ln(\text{effort}) = 1.863 + 0.631 \times \ln(\text{max team size}) + 0.733 \times \ln(fp) + 0.740 \times \text{devplat1} + 0.505 \times \text{devplat2}$$

Devplat1 and devplat2 are dummy variables created for the variable development platform and represent the values midrange and mainframe, respectively (see also Table 2). Table 4 summarizes the results obtained when applying OLS regression and ACE. When applying ACE, analogues were identified using the 145 ISBSG projects as possible source projects for estimating each target project from the Megatec data set. The obtained $R^2$-value for the regression model was 0.82.

The mean and median MRE are in general very high for predictions based on the ISBSG projects. Using OLS regression derives the lowest value. The overall accuracy of ACE when using one analogue is affected by outlying predictions as indicated in the high mean and median MRE values for ACE-1 no SA and ACE-1 with SA. Applying ACE using two analogues significantly decreases the mean MRE values. Many outliers in the ISBSG data are again responsible for the high differences between mean and median MRE. The ISBSG repository is a very heterogeneous data set especially in terms of effort and system size, which makes it difficult to derive accurate predictions.

### 4.3. Comparisons

To address our first question (see Section 1), "What is the difference in accuracy between a traditional technique such as ordinary least-squares regression and Analogy-based estimation?", we compared each of the technique's accuracy (1) for estimates derived from the Megatec data, and (2) for estimates derived from the ISBSG data.

Table 5 reports the *p*-values obtained from a Wilcoxon signed rank test comparing the mean of the two samples. The column "Estimates based on Megatec" reports the results using the Megatec data set (see also Table 3). No significant differences between the techniques can be observed when models are built based on the company-specific data. This result is consistent with the study of the Laturi database, where also no significant differences could be found among various applied modeling techniques using company-specific data [1].

Using multi-organizational data and applying the derived models to Megatec projects (column "Estimates based on

Table 5
Comparison of techniques (*p*-values from Wilcoxon signed rank test)

| | Estimates based on Megatec | | | | Estimates based on ISBSG | | | |
|---|---|---|---|---|---|---|---|---|
| | ACE-1 no SA | ACE-2 no SA | ACE-1 with SA | ACE-2 with SA | ACE-1 no SA | ACE-2 no SA | ACE-1 with SA | ACE-2 with SA |
| ACE-1 no SA | – | | | | – | | | |
| ACE-2 no SA | 0.778 | – | | | 0.003 | – | | |
| ACE-1 with SA | 0.468 | 0.421 | – | | 0.147 | 0.064 | – | |
| ACE-2 with SA | 0.198 | 0.091 | 0.573 | – | 0.005 | 0.717 | 0.020 | – |
| OLS | 0.113 | 0.080 | 0.953 | 0.595 | 0.001 | 0.008 | 0.003 | 0.011 |

ISBSG”), OLS performs significantly better than any variant of ACE. This is also in line with the results from the Laturi, as well as from the ESA study [1,2]. From these results, it seems that using simple OLS regression provides the most accurate results.

Having a closer look at the different ACE variants, we observe significant differences among some of the variants for models based on multi-company data. In this context, size adjustment does not significantly improve the estimates. However, it seems that the use of more than one analogue is a driving factor of significant accuracy improvement. We have evidence that these results are a reflection of the non-linear relationship between system size and effort within the ISBSG data set and the wide range of these project characteristics (see Tables 1 and 2). Therefore, it is much more likely that a selected analogue of the ISBSG data set will be very different in terms of effort from the target project although system size and other project characteristics agree with the target project. For the Megatec data that is more homogenous it is less likely. In this context, the ACE estimates benefit in some instances from the use of the size adjustment algorithm. For the Megatec based comparisons, we see that only ACE using two analogues with size adjustment gets close to a significantly higher accuracy than ACE using two analogues and no size adjustment ($p$-value = 0.091).

Addressing our second question, “Is there a difference between estimates derived from multi-company data and estimates derived from company-specific data?”, Table 6 compares the model accuracy for each technique (the mean MRE values are provided). For almost each ACE variant and OLS regression, significantly different (more accurate) models could be built based on company-specific data than based on multi-company data. This trend is what one would expect, because of the higher homogeneity of the underlying company-specific data set. Moreover, the average productivity of Megatec projects is higher than for ISBSG projects (Table 1). This may explain a consistent overestimation of Megatec projects, when predictions are based on ISBSG projects. Selecting ISBSG analogues of project size similar to the targeted Megatec projects lead to overestimate the effort.

## 5. Discussion and conclusions

To summarize, we can conclude that for Megatec: (1) Generally estimates using their own data are in general much more accurate than using the ISBSG repository. (2) Using their own data, both OLS regression and Analogy should be considered for predictions of new Megatec projects.

Analogy-based estimates are slightly improved on average (but not significantly) when adjusted for the expected size of the target project.

For the ISBSG repository we can conclude: (1) The estimation accuracy when predicting Effort for a non-contributing company (Megatec) of the repository is low, especially when using Analogy-based estimation techniques. (2) Nevertheless, if there is a need to predict Effort by using the ISBSG repository for a non-contributing company, OLS regression should be considered rather than Analogy. Applying other estimation techniques or combinations of techniques [1,2] could be helpful as well, but this was not further investigated in this study.

The difference in mean MRE (Table 6) for OLS

Table 6
Megatec vs. ISBSG based estimates (Wilcoxon signed rank test)

| Comparison | Mean MRE for Megatec based estimates (see Table 3) | vs. | Mean MRE for ISBSG based estimates (see Table 4) | Differences in mean MRE | *p*-value |
|---|---|---|---|---|---|
| ACE-1 no SA | 0.63 | vs. | 2.48 | 1.85 | 0.016 |
| ACE-2 no SA | 0.54 | vs. | 1.47 | 0.97 | 0.058 |
| ACE-1 with SA | 0.38 | vs. | 2.39 | 1.99 | 0.000 |
| ACE-2 with SA | 0.37 | vs. | 1.43 | 1.06 | 0.003 |
| OLS | 0.37 | vs. | 0.61 | 0.24 | 0.014 |

regression is much smaller than for the ACE variants. This leads to the conclusion that OLS regression seems to be a more robust technique than Analogy. Using Analogy, we found a consistent overestimation of the Megatec target projects and high MRE values. Furthermore, it is completely logical that size adjustment needs to be applied on the Analogy-based estimates. The regression model-based estimates already include a size adjustment by default since system size is one of the independent variables in the model.

It was easy to select variables for this analysis, as they had to be available in both data sets as well as having an influence on effort. The same variables, however, do not necessarily have the same measured impacts for both data sets. Thus, the difficulty in linking the cost relationships from Megatec to ISBSG and vice versa may have resulted in poor estimates.

It is also worth investigating whether the size adjustment should be a linear or perhaps a different form. This again depends on whether the data sets are comparable in terms of variables and distributions.

The practical implications of these results are: (1) Before using a repository for cost estimation the company should collect the same or at least similar variables and it should ensure that the cost drivers are comparable. (2) Given that OLS and Analogy based estimates had similar accuracy within the company, if an organization were to have a fairly sizeable data set and a very good understanding of their own cost drivers, they may prefer to use Analogy as their estimation method. In our experience we have found that practitioners are more comfortable with the Analogy method and prefer to use this technique in informal estimation procedures. Comparison of estimates derived in this way with OLS model estimates would also be worthwhile. (3) If an organization does not have a sizeable data set of their own, then a regression model derived from public data sets would be more advisable than using Analogy. Cost factors derived from such regression modeling should be checked with conventional wisdom within the organization in order to provide informal model validation.

In this paper, we compared one parametric and one non-parametric cost estimation technique (OLS regression and ACE, respectively). In order to derive more generalizable conclusions, future work remains to be done evaluating a variety of other cost estimation techniques on those two data sets. In addition, an investigation of consistencies and differences in the results of previous studies [1,2] is intended. This also implies an in-depth investigation of the characteristics of the data sets.

## Acknowledgements

This work was supported by grants from Megatec, the Fraunhofer Institute for Experimental Software Engineering, the Centre for Advanced Empirical Software Research (CAESAR) at UNSW, and the CSIRO. Thanks also go to the International Software Benchmarking Standards Group (ISBSG) for the repository used in the study.

## References

[1] L.C. Briand, K. El Emam, K. Maxwell, D. Surmann, I. Wieczorek, An Assessment and Comparison of Common Software Cost Estimation Models. In: Proceedings of the 21st International Conference on Software Engineering, ICSE 99, Los Angeles, USA 1999, pp. 313–322.

[2] L.C. Briand, T. Langley, I. Wieczorek, A replicated Assessment of Common Software Cost Estimation Techniques. In: Proceedings of the 22nd International Conference on Software Engineering, ICSE 2000, Limerick, 2000, pp. 377–386.

[3] S.D. Conte, H.E. Dunsmore, V.Y. Shen, Software Engineering Metrics and Models, Benjamin/Cummings, Menlo Park, CA, 1986.

[4] S.J. Delany, P. Cunningham, W. Wilke, The limits of CBR in software project estimation, Proceedings of the 6th German Workshop on Case-Based-Reasoning, Berlin, 1998.

[5] G.R. Finnie, G.E. Wittig, J.-M. Desharnais, A comparison of software effort estimation techniques: using function points with neural networks, case-based reasoning and regression models, Journal of Systems and Software 39 (3) (2000) 281–289.

[6] A.R. Gray, S.G. MacDonell, A comparison of techniques for developing predictive models of software metrics, Information and Software Technology 39 (2000) 425–437.

[7] M. Hardy, Regression with Dummy Variables, Sage Publications, Beverley Hills, CA, 1993.

[8] R. Jeffery, J. Stathis, Function point sizing: structure, validity and applicability, Empirical Software Engineering (1996) 11–30.

[9] C.J. Lokan, An empirical study of the correlations between function point elements, Proceedings of the 6th International Software Metrics Symposium, 1999, pp. 200–206.

[10] K. Maxwell, L. Van Wassenhove, S. Dutta, Performance evaluation of general and company specific models in software development effort estimation, Management Science June (1999).

[11] K. Maxwell, P. Forselius, Benchmarking Software Development Productivity, IEEE Software January/February (2000).

[12] T. Mukhopadhyay, S.S. Vicinanza, M.J. Prietula, Examining the feasibility of a Case-Based reasoning model for software effort estimation, MIS Quarterly June (1992) 155–171.

[13] M. Ruhe, Comparative Study of Project Effort Estimation Methods by Using Public Domain Multi-Organizational and Organization-specific Project Data. CAESAR-Report 99/4 (www.caesar.unsw.edu.au).

[14] J.A. Rice, Mathematical Statistics and Data Analysis, 2nd ed., Duxbury Press, 1995.

[15] L. Schroeder, D. Sjoquist, P. Stephan, Understanding Regression Analysis: An Introductory Guide, Quantitative Applications in the Social Sciences, Sage Publications, Newbury Park, USA, 1986.

[16] M. Shepperd, C. Schofield, Estimating software project effort using analogies, IEEE Transactions on Software Engineering 23 (12) (2000) 736–743.

[17] Software Project Estimation: ISBSG A Workbook for Macro-Estimation of Software Development Effort and Duration, 1999.

[18] F. Walkerden, R. Jeffery, An empirical study of analogy-based software effort estimation, Empirical Software Engineering 42 (1999) 135–158.

[19] S.M. Weiss, C.A. Kulikowski, Computer Systems that Learn, Morgan Kaufmann, San Francisco, USA, 1991.