Effort and Cost in Software Engineering: A Comparison of Two Industrial Data Sets

Hennie Huijgens, Arie van Deursen
Delft University of Technology
Delft, The Netherlands
h.k.m.huijgens@tudelft.nl
arie.vandeursen@tudelft.nl

Leandro L. Minku
Department of Informatics
University of Leicester, UK
leandro.minku@leicester.ac.uk

Chris Lokan
University of New South Wales
Canberra, Australia
c.lokan@adfa.edu.au

ABSTRACT

Context: The research literature on software development projects usually assumes that effort is a good proxy for cost. Practice, however, suggests that there are circumstances in which costs and effort should be distinguished. *Objectives*: We determine similarities and differences between size, effort, cost, duration, and number of defects of software projects. *Method*: We compare two established repositories (ISBSG and EBSPM) comprising almost 700 projects from industry. *Results*: We demonstrate a (log)-linear relation between cost on the one hand, and size, duration and number of defects on the other. This justifies conducting linear regression for cost. We establish that ISBSG is substantially different from EBSPM, in terms of cost (cheaper) and duration (faster), and the relation between cost and effort. We show that while in ISBSG effort is the most important cost factor, this is not the case in other repositories, such as EBSPM in which size is the dominant factor. *Conclusion*: Practitioners and researchers alike should be cautious when drawing conclusions from a single repository.

CCS CONCEPTS

• **General and reference** → Cross-computing tools and techniques → Metrics

KEYWORDS

Software Economics; Evidence-Based Software Portfolio Management; EBSPM; Benchmarking; ISBSG; Cost Prediction.

ACM Reference format:

H. Huijgens, A. van Deursen, L. Minku, and C. Lokan. 2017. Effort versus Cost in Software Development: A Comparison of Two Industrial Data Sets. In *ACM Proceedings of the 21th International Conference on Evaluation and Assessment in Software Engineering,* 10 pages. EASE'17, June 15-16, 2017, Karlskrona, Sweden. © 2017 Association for Computing Machinery.
ACM ISBN 978-1-4503-4804-1/17/06...$15.00
http://dx.doi.org/10.1145/3084226.3084249

1 INTRODUCTION

A good understanding of the cost of software development, that is *the real cost that companies pay* for their software development activities, is important for business to make better decisions [1] [2]. Two issues arise with regard to cost of software projects.

First, cost and effort are often looked upon as equivalent. At the best effort is assumed to be a good proxy for cost, where the emphasis seems to be more on effort, and less on cost. Frequently studies are found that claim to be about cost estimation, yet the study itself analyzes effort as the main subject, e.g. [3] [4] [5] [6]. To judge the evidence of cost-savings in global software engineering, Šmite et al. [1] reviewed more than five hundred articles on global software engineering, and found that only fourteen articles presented evidence of cost-savings [1].

Second, many benchmarks are available for software projects. Jones [7] mentions no-less than twenty-five sources of software benchmarks. Menzies and Zimmermann [8] inventoried thirteen repositories of software engineering data. Yet, cost data are missing in most of them. One might expect that the idea of cost as an important factor for evidence-based steering on software engineering activities, combined with the availability of many sources for benchmarking, would lead to substantial knowledge about efficiency in terms of cost. Yet, in practice this is not the case. To illustrate this, a recent study of Bala and Abran [9] on how to deal with missing data in the repository that is maintained by the International Software Benchmark Standards Group (ISBSG) [10], does not mention cost once, while a large part of the study is about effort related issues.

Jørgensen and Shepperd's systematic review of software development cost estimation studies [11] states that the "main cost driver in software development projects is typically the effort and we, in line with the majority of other researchers in this field, use the terms cost and effort interchangeably in this paper." Existing literature assumes that cost and effort are the 'same thing'. However, from interactions with industry, we believe the relation between both metrics is not that simple. The collection of these metrics needs to be done in different ways, each with their own difficulties. Furthermore, both metrics may be affected by different variables, such as country, inflation, and commercial aspects. While cost data might be reliable from an accounting viewpoint, they might include different actual data across projects phases. In order to emphasize the importance of cost, especially top-level decision makers highly value cost transparency, as we found in a study on pricing of software projects [12]. To explore the differences and commonalities between cost and effort, we use two large software repositories; the EBSPM repository [13] [14] [15] and the ISBSG repository [10], with the objective of gaining insights into the usefulness of historic effort and cost data for benchmarking and cost estimation purposes. This is one of the first studies in which EBSPM is used in comparison with other benchmark datasets. We perform our analysis as an exploratory study based on two data sets. Therefore, our datasets and results may not generalize. Our objectives are to (1) determine the similarities and differences between cost and effort, (2) determine whether these make cost modelling a substantially different problem from effort modelling, and (3) discuss potential reasons for the differences. We address the following research question:

RQ1 *How do the EBSPM-repository and the ISBSG-repository compare with regard to the size, effort, cost, duration, and number of defects of software projects?*

When building the EBSPM-repository we experienced that effort and cost are not the same, mainly due to complex interactions with industry. A software company's project portfolio is usually built from differently organized cost structures. Simple structures like effort times hourly tariff are mixed with many other factors such as upfront agreed fixed price activities, delivery strategies such as agile (Scrum) where teams are budgeted for a period of one year or longer, sourcing strategies with globally distributed teams, and many more. Due to this the theory that cost can simply be calculated out of effort based on hourly rates is spurious, and does not hold for many software projects in industry.

We experienced in practice, that many decision makers in industry use cost as a major indicator for their decisions, and not effort as such. The interactions mentioned above make us believe that effort is not a simple proxy for cost, and that both metrics should be looked upon in research as an autonomous subject.

This paper is structured as follows. In Section 2 we describe our research approach. Section 3 is about the results of our research. In Section 4 we discuss the outcomes and implications and threats to validity. In Section 5 we link these with related work. Finally, in Section 6 we describe conclusions.

2 RESEARCH APPROACH

For this exploratory study we apply a qualitative and a quantitative approach. We discuss differences between cost and effort, based on our previous interactions with industry, and we analyze correlations between cost and effort based on a subset of the ISBSG-repository. We create cost models based on two data sets; a subset of the ISBSG-repository and the EBSPM-repository, and we investigate whether typical results, such as influence of size and differences between data sets, obtained by the literature on software effort estimation also hold for cost.

We test whether both repositories are different from each other and in what degree they are fit for use to build a prediction model for effort and for cost. In particular we examine what independent variables are relevant for predicting cost. In both cases we look for the best fit; meaning that for each dataset different prediction models can apply. Furthermore, we evaluate the performance of both repositories from the perspective of a full software portfolio, by analyzing specific samples of software project data against the content of the EBSPM-repository [13]. Finally, we look for causes that might explain our findings, by studying existing literature on the subject of effort and cost of software projects.

2.1 The EBSPM-repository

The EBSPM-repository is a collection of data from approximately five hundred finalized software projects. Data is collected by specialized measurement teams within three different companies (banking and telecom) in The Netherlands and Belgium. Data is available for projects of different business domains. The functional size of all projects is measured in function points by specialized function point analysts. We focus on size, effort, cost, duration, and defects, which are all present in this data set, although effort is only collected for a limited number of 22 out of 488 projects. All software projects have been performed and measured in the period of nine years from 2008 to 2016. An important feature of the EBSPM-repository is that it contains data of a company's software portfolio as a whole, representing a variety of projects, business domains, delivery approaches, and sourcing strategies. Where other repositories – like ISBSG – focus on projects as such, EBSPM focus on portfolios instead. This enables us to analyze good practice versus bad practice projects from a portfolio point of view [14]. In order to reduce effects of inflation we calculate all Euro values in the EBSPM-repository that we used for our study to the 2015 value, based on the calculation tables of the International Institute of Social History[1]. Table 1 shows descriptive

Table 1. Descriptive Statistics of the EBSPM Project Data.

n = 488	Size (FPs)	Cost (Euros)	Effort (Hours)	Duration (Months)	Nr. of Defects
Minimum	4.00	329	31	0.90	0.00
First Quartile	38.25	71684	99	5.39	6.75
Median	115.50	293166	807	8.41	20.50
Mean	216.07	637935	3202	8.96	70.09
Third	248.75	740559	2391	11.09	55.25
Maximum	4600.00	7523527	22096	26.84	1586.00
Skewness	6.26	3.68	466 NAs	0.96	222 NAs

NAs indicate fields with no data available for effort and number of defects; due to this skewness could not be calculated. We emphasize that due to 466 NAs with regard to Effort only 22 projects are included in the calculations on Effort.

[1] http://www.iisg.nl/hpw/calculate.php

statistics of the EBSPM-repository. The repository and the accompanying tool are described in more detail in a separate tool description [13].

2.2 The ISBSG-repository

ISBSG is a repository collected by the International Software Benchmark Standards Group (ISBSG) [10], that is licensed to software companies that wish to use their tools for estimation and benchmarking purposes (for researchers ISBSG data is available free of charge). For the purpose of this study we use ISBSG version D&E Corporate Release 17 April 2013 [10]. The full ISBSG-repository consists of 6010 projects; we select those projects for which there is data on cost, size, and duration:

1. We exclude all projects with no size recorded (*Functional Size* is empty).
2. We solely include projects that are counted in function points (values "IFPUG 4+", or "NESMA" in *Count Approach*).
3. We exclude all other projects. We exclude projects with no cost recorded (*Total Project Cost* is empty).
4. We exclude projects with no project duration recorded (*Project Elapsed Time* is empty).
5. Finally, we exclude all projects that were executed before the year 2000 (*Year of Project*), in order to limit the subset of projects to periods close to those in the EBSPM-repository.

After filtering, a subset of 172 projects is available for further comparison of effort and cost in the EBSPM-repository. In order to normalize all project cost in the subset, we convert the data in *Project Cost* to Euros based on historical exchange rates as denoted by trading company Oanda[2]. To do so we select the 1st of January of the applicable *Year of Project* in ISBSG as begin-date and the 31st of December of the *Year of Project* from ISBSG as end-date. Alike the EBSPM-repository we finally calculate all Euro values to the 2015 value, in order to reduce any effects of inflation.

Table 2. Descriptive statistics of the ISBSG project data.

n = 172	Size (FPs)	Cost (Euros)	Effort (Hours)	Duration (Months)	Nr. of Defects
Minimum	11.00	1627	183	1.00	0.00
First Quartile	40.50	23873	497	4.00	7,00
Median	125.00	68974	1445	10.00	13.00
Mean	307.40	180813	3890	11.31	98.85
Third	296.00	215861	3760	17.60	25.00
Maximum	10571.00	1915823	70035	54.00	2395.00
Skewness	9.90	2.96	5.80	1.02	(79 NAs)

Table 2 gives an overview of descriptive statistics of the ISBSG-subset that we used in this study. An comprehensive overview of the ISBSG dataset that we used, including the calculated cost data, is to be found in a Technical Report [16].

2.3 Analysis Procedure

We perform a series of statistical tests to examine whether both repositories are significantly different. We perform Wilcoxon rank sum tests to compare differences between size, cost, effort, duration, and number of defects, and differences per size. To check for normality, we perform Shapiro tests and analyze histograms. To examine the prediction power of both datasets we use linear regression, stepwise linear regression, and CART trees, the latter as suggested in [17]. For linear regression, we also eliminate influential observations based on Cook's distance. Besides that, we compare both repositories by mapping the project data of the ISBSG subset on the Cost Duration Matrix in the EBSPM-tool [13], analyzing the following performance indicators:

1. *Cost per Function Point* (FP): the weighted average cost (in Euros) per FP, where size (FP) is the weighting factor, instead of number of projects.
2. Duration *per FP*: the weighted average duration (in calendar days) that it took to deliver a FP.
3. *Number of Defects per FP*: the weighted average defects (from system integration test to technical go live) per FP.

We compare the performance of projects in the EBSPM-repository as a whole with ISBSG data. Within the scope of this study we do not look at company-specific causes; we look at 650 projects, and assume that company specific causes are not significant for common trends, as indicated in previous research [14]. Company-specific factors do not necessarily give us homogeneous information about aspects like country or sourcing strategy, because companies may be spread through different countries, their projects may be very heterogeneous in terms of sourcing strategy.

3 RESULTS

In order to examine the fitness of both repositories to build a prediction model for effort and cost, we performed a series of statistical tests. A detailed overview of the results of these tests is included in a Technical Report [16]. In this section we provide a summary of the most relevant outcomes.

In accordance with what is known from related work [3] [18] [19], we found that size, cost, and to a lesser extent duration, in both datasets were not normally distributed, indicated by a relatively high score for skewness (see Table 1 and Table 2). Boxplots of both repositories (see Figure 1) confirm this observation. We also checked for normality by performing Normality Shapiro tests. These result in violations for all numerical variables (all p-values were smaller than 2.4e-10; see the Technical Report [16]). To examine differences between both datasets we performed Wilcoxon ranked sum tests with Holm-Bonferroni corrections to compare overall differences, and differences per size (see Table 3). P-values indicate that overall Cost and Cost per Size are significantly different, whereas the other metrics are not significantly different.

[2] http://www.oanda.com/currency/average.

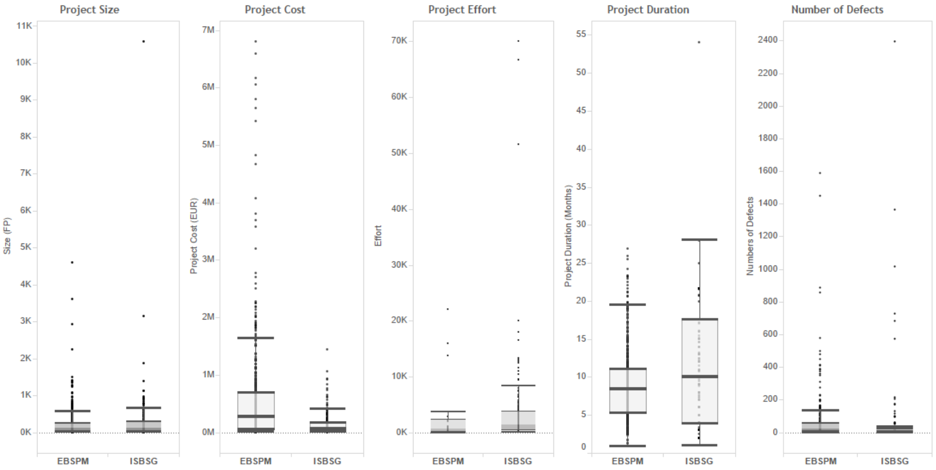


Figure 1 Boxplots showing the differences between the projects in the EBSPM-repository and the ISBSG-subset.

3.1 Linear Regression

In order to further examine differences between both datasets, we use residuals versus fits and QQ plots to examine the points with values that are substantially larger than the rest. We perform several tests in order to examine which model fits best based on both datasets. As the data are not normal, we apply a log transformation, as recommended in the literature [20].

Table 3. Results from the Wilcoxon rank sum tests comparing EBSPM and ISBSG.

	Median EBSPM	Median ISBSG	W	p-value
Size	115.50	125.00	39063	0.1539
Cost	293166	68974	60446	0.0000*
Effort	807	1445	1369	0.0351
Duration	8.41	10.00	36768	0.0128
Number of Defects	20.50	13.00	13441	0.2592
Cost / Size	2684	602	72157	0.0000*
Effort / Size	NA	13.13	1668	0.3674
Duration / Size	2.09	1.62	46123	0.0650
Number of Defects / Size	0.18	0.31	11024	0.0974

The highlighted and with an asterix marked rows indicate statistically significant difference when applying Holm-Bonferroni corrections based on 7 comparisons, at the overall level of significance of 0.05; due to this correction Effort and Duration are not assessed significantly different.

Table 4. Improvements in fit (Multiple R-squared).

	EBSPM	ISBSG
Linear regression, without log	0.7663	0.1416
Linear regression, with log	0.7012	0.6815
Linear regression, no influential observations	0.8123	0.9029
Linear regression, with additional factors[1]	0.8839	0.9226

[1] For EBSPM the factors Organization, Business Domain, Development Approach, and Year Go Live were added. For ISBSG the factor Development Type was added. Effort was not included in any of the models above.

A subset of the plots is shown in Figure 2; indicating the differences between the EBSPM-repository and the ISBSG-subset. The upper left plot (EBSPM) and the upper right plot (ISBSG) give an idea of whether the relationship between dependent and independent variables is linear. When the red lines are horizontal, the relationship is likely to be linear. The plots indicate that for both EBSPM and ISBSG the relationship deviate a bit from linearity, but these are small. Some violations to homoscedasticity (homogeneity of variance) are indicated, especially for ISBSG, as indicated by the plot at the bottom right.

Both QQ-plots at the bottom left (EBSPM) and bottom right (ISBSG) indicate fairly normal residuals for EBSPM. However, for ISBSG, the distribution seems less normal; indicating that linear regression is less adequate for ISBSG than for EBSPM. Application of backward stepwise linear regression results in a similar fit as linear regression. We determined highly influential observations based on cook's distance and the stepwise model [21]; removing these in the stepwise model had no effect (see Table 4).

In Table 4 we show how much improvement in fit (multiple R-squared) we gain by using log, even though log is not always making things normal. We use linear (not stepwise) regression, so that we can know the estimate of the slope and result of the statistical tests of the hypothesis that the slope equals zero, with respect to all independent variables. Note that effort is not included in the

Table 5. Results of fitting a linear model.

	EBSPM		ISBSG	
	Estimate	Star-rate	Estimate	Star-rate
(Intercept)	8.11367	***	5.53994	***
Log(Size)	0.58485	***	0.36232	**
Log(Duration)	0.35923	***	0.04552	
Log(Effort)	NA	NA	0.44362	**
Number of Defects	0.28517	***	0.09518	

Results of fitting a linear model, with log transformation in R. Highly influential observations are removed before the model was generated.

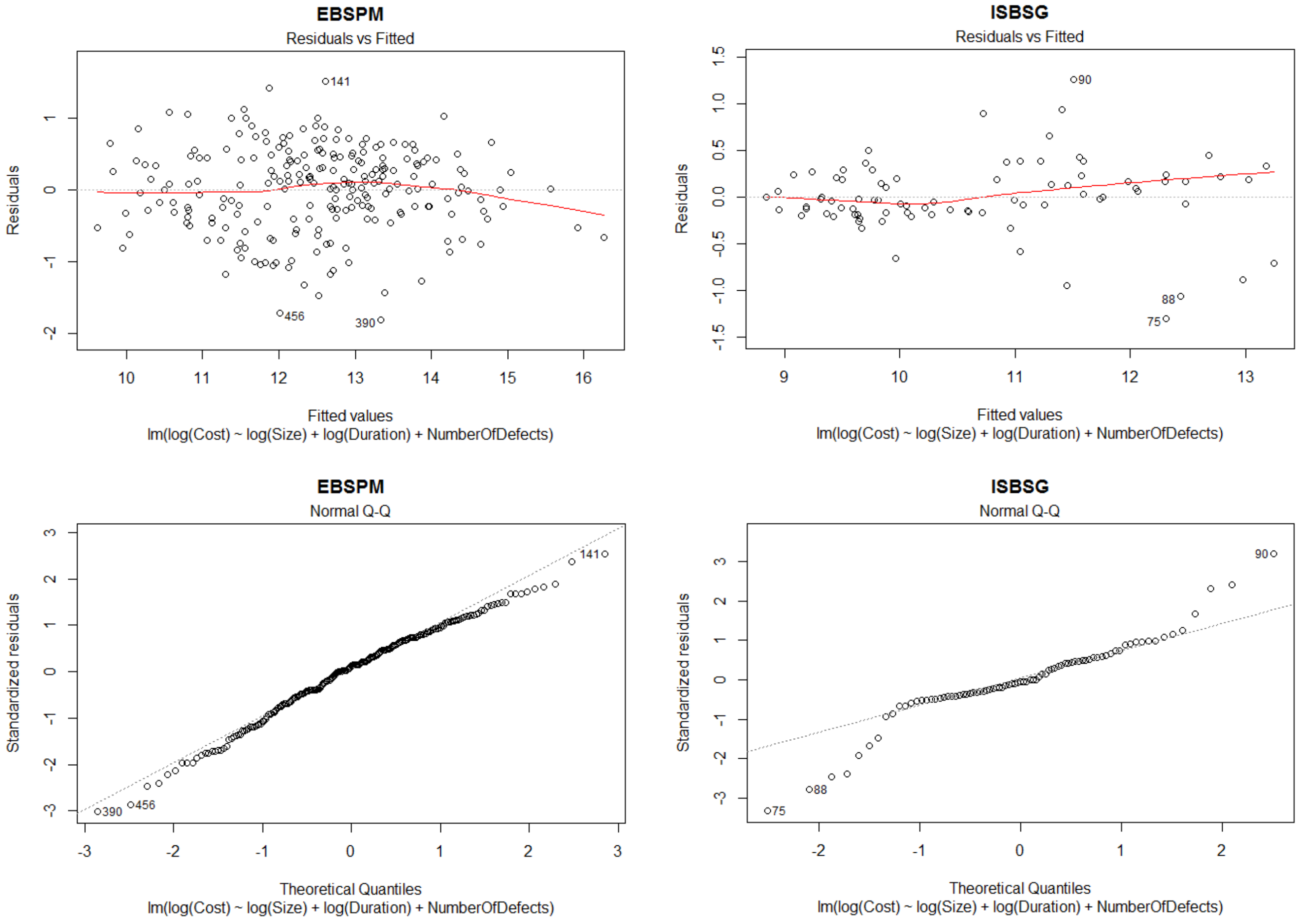


Figure 2. Comparison between Residuals and Fitted Values of the EBSPM-repository (top left) and the ISBSG-subset (top right), and QQ-plots of the EBSPM-repository (bottom left) and the ISBSG-subset (bottom right). The plots are generated on log(Size), log(Duration), and Number of Defects. Highly influential observations are removed from both datasets before the plots where generated. In all plots no log is applied for Number of Defects.

models; in terms of linear regression, excluding effort leads to a better R-squared (0.9029), than including effort (0.7159). Our results indicate that for EBSPM adding effort does not help with linear regression. Fitting a linear model, with log transformation, where highly influential observations are removed before the model was generated, shows that in the EBSPM-repository Size, Duration, and Number of Defects are significantly related to Cost (see Table 5). This confirms common knowledge from related work that Size is a strong predictor of Cost. In addition, it shows that for EBSPM Duration and Number of Defects are strong predictors for Cost too. A similar test on the ISBSG-subset, shows that Size and Effort are relevant factors; however, Duration and Number of Defects are not. A warning is in place here; these results might be influenced by the fact that for EBSPM not enough effort data was available to fit a linear model (see the NAs in Table 5).

3.2 Regression Trees

Given the potential violations to the assumptions of linear regression (especially when using ISBSG), we created regression trees for both EBSPM and ISBSG, to see if they agree with our conclusions on linear regression. When examining the regression tree based on the EBSPM-repository (see Figure 3, left), we observe that Size is the root node. This indicates that this is the most important attribute for predicting Cost. Duration appears for the first time at the third level of the tree, indicating that Duration is less important than Size, but still relevant for predicting Cost. Number of Defects does not appear in the regression tree; it is considered as irrelevant for predicting Cost.

Examining the regression tree of the ISBSG-subset (see Figure 3, right), we find that Cost and Effort do have a strong relationship, as suggested in existing literature. Duration and Number of

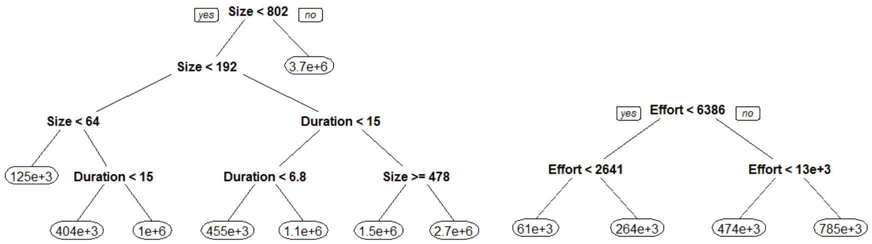


Figure 3. Regression Trees based on the EBSPM-repository (left) and the ISBSG-subset (right). In the EBSPM dataset not enough effort data was available to perform a regression tree with effort included.

Defects are absent, indicating that these are not relevant for the purpose of estimating Cost for ISBSG.

By aggregating the findings obtained by linear regression and regression trees, we can make the following observations. For EBSPM, both Size and Duration are very relevant, with Size being more relevant than Duration. This is reflected both in the linear regression and regression tree. Number of Defects are less important and may or may not be relevant, given their smaller coefficient in linear regression and absence in the regression tree. For ISBSG, Size and Effort are relevant and Duration and Number of Defects are not. This is reflected in the linear regression and partly in the regression tree.

3.3 Mapping on the EBSPM-tool

To visualize performance in terms of size, cost, duration, and defects, we mapped the subset of 172 ISBSG projects on the content of the EBSPM-repository, by using the EBSPM-tool [13] (see Figure 4). To analyze the overall performance of both repositories, we calculated three Performance Indicators (see Table 6).

Table 6. Overview of Performance Indicators.

	EBSPM	ISBSG
Observations	488	172
Cost (Euros) per FP	3,630	795
Duration (Calendar Days) per FP	5.53	6.74
Number of Defects per FP	0.35	0.35

All performance indicators are calculated as weighted average, with size as weighting factor (instead of number of projects).

Overall average Duration per FP as measured in the EBSPM-repository (5.53) does not match ISBSG Duration per FP (6.74), but differences are relatively small. The trendline for duration of the ISBSG-subset (the horizontal red line) is 23% below the EBSPM-trend (the horizontal dotted line), indicating that ISBSG projects on average took 23% longer to finalize than the EBSPM ones. On quality no differences occur, as both datasets show a weighted average of 0.35 Number of Defects per FP. The median Number of Defects is statistically similar according to the Wilcoxon tests.

Yet, differences on cost are huge. Average Cost per FP in the EBSPM-repository is 3,630 Euros, while the ISBSG repository shows an average of 795 Cost per FP. The ISBSG trendline in Figure 4 (the vertical red line) is as much as 80% to the right of the EBSPM-trend (the vertical dotted line), indicating that ISBSG projects were 80% cheaper in terms of Cost per FP than EBSPM ones. With regard to project size (in function points) we observe that on average ISBSG projects show a size of 307 FPs, where the EBSPM projects show an average size of 216 FPs. However, these relatively small differences in size do not explain the huge difference in cost in both datasets. This is confirmed by the Wilcoxon Ranked Sum Test with Holm-Bonferroni corrections, which reveal no significant differences in size between EBSPM and ISBSG.

3.4 Key Findings

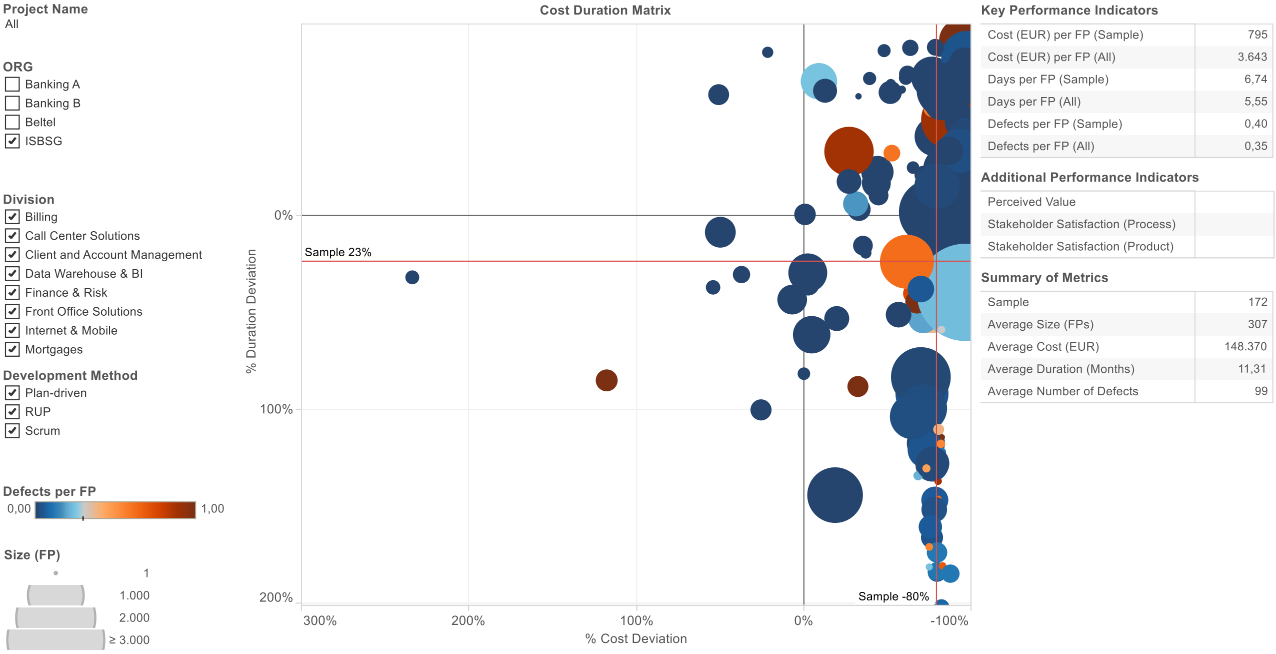
Looking at our research question:

RQ1 How do the EBSPM-repository and the ISBSG-repository compare with regard to the size, effort, cost, duration, and number of defects of software projects?

we determine the following key findings in this study: (1) We demonstrate a (log)-linear relation between cost on the one hand, and size, duration and number of defects on the other (as illustrated in Figure 2). This justifies conducting linear regression for cost. (2) We establish that ISBSG is substantially different from, e.g., EBSPM, in terms of cost (cheaper) and duration (slower), and the relation between cost and effort. This implies that practitioners and researchers alike should be cautious when drawing conclusions from a single repository. (3) We show that while in ISBSG effort is the most important cost factor, this is not the case in other repositories, such as EBSPM in which size is the dominant factor.

4 DISCUSSION

The main questions that arise from the analysis that we performed, is whether the large differences that we found in Cost, and the fact that Duration and Number of Defects are of influence to Cost in the EBSPM-repository, and not in the ISBSG-subset, can be explained in any way?



Project Name
All

ORG
□ Banking A
□ Banking B
□ Beltel
☑ ISBSG

Division
☑ Billing
☑ Call Center Solutions
☑ Client and Account Management
☑ Data Warehouse & BI
☑ Finance & Risk
☑ Front Office Solutions
☑ Internet & Mobile
☑ Mortgages

Development Method
☑ Plan-driven
☑ RUP
☑ Scrum

Key Performance Indicators

Cost (EUR) per FP (Sample)	795
Cost (EUR) per FP (All)	3.643
Days per FP (Sample)	6,74
Days per FP (All)	5,55
Defects per FP (Sample)	0,40
Defects per FP (All)	0,35

Additional Performance Indicators

Perceived Value	
Stakeholder Satisfaction (Process)	
Stakeholder Satisfaction (Product)	

Summary of Metrics

Sample	172
Average Size (FPs)	307
Average Cost (EUR)	148.370
Average Duration (Months)	11,31
Average Number of Defects	99

Figure 4. The Cost Duration Matrix in the EBSPM-tool showing a sample of 172 projects from the ISBSG repository, plotted against 488 projects from the EBSPM-software project repository. The ISBSG data points are plotted as circles, the EBSPM is represented as 0%-dotted lines. The size of the data points indicates the size of a software project (bigger circles indicate bigger projects in FPs). The color of the data points indicates the quality (redder circles indicate more defects per FP). The dotted black lines indicate the average of the whole EBSPM-repository for cost (vertical line) and duration (horizontal line). The red lines indicate the average for the sample selected (in the figure the ISBSG-subset) for cost (vertical line) and duration (horizontal line). The analysis indicates that the ISBSG-projects performed on average 80% cheaper, but 23% slower than the projects in the EBSPM-repository. This figure was also used in a description of the EBSPM-tool [12].

Our analysis shows that the EBSPM-repository and the ISBSG-subset are significantly different with regard to cost of finalized projects. Looking at the significant differences in the EBSPM-repository between different companies, as shown in the boxplots in Figure 1, we tend to agree with the idea that building a dedicated repository of historic projects (a single-company model) helps companies to make better predictions of new projects and to improve benchmarking and analysis of ongoing and finalized projects. Also the idea of clustering fits with our experience that some business domains (e.g. data warehouse) show cost patterns that deviate from general ones, and therefore need to be looked upon in a specific way.

Based on the outcomes of our comparison, we argue that, effort and cost are interrelated, at least in the ISBSG-subset that we studied. However, when plotted over time both metrics seem to show a more complex relationship (see the Technical Report [16] for a figure on development of project metrics over time). Besides the common idea that cost reflects effort times hourly tariff, many other factors play a role here: e.g. market issues, productivity changes over time, and sourcing strategies. However, no major cost per FP changes are to be found in ISBSG data when looked upon over time. What strikes in our study, however, is the remarkable difference in Average Weighted Cost per FP between both studied repositories, as depicted in Table 6; in the ISBSG-subset development of one FP cost on average 795 euro, while in the EBSPM-repository 3630 euros are needed to do so.

The big differences that we found between project cost in both repositories could not be explained based on Size, Duration and Number of Defects of the software projects. Unfortunately, no relevant Effort data was available in the EBSPM-repository, so we cannot conclude anything about that. In earlier research [14] we found indications that might be of influence to cost of projects, such as software delivery strategies (agile and release-based, steady heartbeat, fixed teams) and specific business domains. Unfortunately, the ISBSG descriptions don’t tell us much about these aspects. A key factor in the comparison of both repositories might be different labor costs in different countries. EBSPM projects come from the Netherlands and Belgium, however a significant part of the projects is performed in cooperation with suppliers in other countries (e.g. India). ISBSG projects come from a wide range of countries, including some whose IT industry is based on labor costs being low. Since ISBSG does not normally release data about the country in which a project was performed, it is hard for users of ISBSG cost data to take this into account. However, when the total cost of the ISBSG projects in the subset that we used are divided by the total effort, this results in an average hourly rate of 43.01 Euro. A complicating factor here might be that in the ISBSG repository, total project effort is a mandatory field [22], where cost is not. In the EBSPM-repository; cost is mandatory, mainly due to the recurring difficulties that the measurement teams involved in EBSPM experienced in collecting reliable effort data in practice.

One major aspect that we assume to be key in this comparison, is that in the EBSPM-approach a software portfolio as a whole of each company is measured, e.g. the good and the bad projects. The ISBSG-approach focuses at single projects, and therefor might not reflect a company's portfolio performance in a holistic way.

4.1 Implications

Taking into account that collecting reliable effort in industry is difficult, and that we found substantial differences between both studied datasets, we point out that researchers and practitioners should take great care to understand the data they are using when making estimates. A remark with regard to using repositories for prediction purposes, is that not only the adequacy of different predictive models for each dataset varies, but also the most relevant independent variables. This is in line with previous research on effort estimation [43]. Care must also be taken when using data from one repository to predict projects for companies not represented in the repository. As we show, projects from different repositories can be considerably different. In particular, we observe that the two repositories had projects with similar duration and number of defects (independent variables), but different costs: relevancy filtering and locality-based approaches, which have been achieving promising results for cross-company effort estimation [44] [45], might not work well for predicting cost.

4.2 Threats to Validity

Our study focuses on differences between size, cost, duration, and number of defects in both repositories. All other factors, such as organization, business unit, primary programming language, development approach, or development type are not included in the analysis. We are aware of the fact that including different factors might influence the outcome of the analysis, yet we'd like to argue in our favor that overall both repositories are a representative subset of any company's software project portfolio. Statistical tests showed no evidence that organization or business domain was of significant influence for cost prediction, where size, duration and numbers of defects actually were. As a remark we mention that related work shows that a relationship between cost and business domain is applicable in the EBSPM-repository [14], and with effort in the ISBSG repository [26].

Two important limitations might be of influence to our study. Only a small part of the software projects in the ISBSG-repository includes cost data; a subset of 172 out of 6010 projects (3%) was applicable for our analysis. On the contrary, only a small part of the EBSPM-repository holds effort data; 22 out of 488 projects (4.5%). We emphasize that our findings are not to be generalized without any restrictions to other software repositories. In order to assure the quality of Function Point counting in the ISBSG-subset; the subset only contains projects with an Unadjusted Function Point Rating 'A' (the unadjusted FP was assessed as being sound with nothing being identified that might affect its integrity) or 'B' (the unadjusted function point count appears sound, but integrity cannot be assured as a single figure was provided).

With regard to quality assurance of the EBSPM-repository: projects were measured by experienced, often certified measurement specialists. Project data was based on formal project administrations and reviewed by stakeholders (e.g. project managers, product owners, finance departments, project support). All projects were reviewed thoroughly by the first author of this study before they were included in the EBSPM-repository. We used the default parameters from the CART package in R to build regression trees. Finally, we emphasize that we used – where possible – a relevant and extended subset of statistical tests to analyze both repositories. Our goal was to link evidence found from one test to confirming results from other tests too.

5 RELATED WORK

5.1 Repositories for Benchmarking

Our findings are not all new; related work confirms large differences between both within-company and cross-company repositories. Much research is performed on whether organizations should use cross-company datasets for estimation and benchmark purposes or whether they should collect their own historic data [23] [24] [25] [26] [27] [28]. The outcomes of studies are mixed.

Garre et al. [29] emphasize that "in the case of large project databases with data coming from heterogeneous sources, a single mathematical model cannot properly capture the diverse nature of the projects". Apparently, other benchmark sources are important. Also the usually "large disparity of their instances" lead to problems [29]. Many researchers have now a better awareness that single companies can themselves have very heterogeneous projects. As a consequence, the community has started to question the usefulness of the distinction between the terms 'cross' and 'within' [30].

Abran et al. [31] mentions a lack of historical data in many companies, as a solution they propose a simulation using the ISBSG-repository. Lokan et al. [32] position ISBSG as a 'low cost initial determination of an organization's industry position and its comparative strengths and weaknesses'. Fernández-Diego et al. [33] inventoried the use of the ISBSG-repository by performing a systematic mapping review on 129 research papers. They found that in 70,5% of the studies prediction of effort is the main focus. In 55% of the papers ISBSG is used as the only support.

Oligny et al. [34] propose a duration prediction model that is based on ISBSG data, that can deliver a 'first order' estimate to project managers. Lokan et al. [32] state that "ISBSG does not claim that the repository represents the whole industry; rather, it believes that the repository only represents the best software companies" [32]. Cheikhi and Abran [35] mention the ISBSG repository and Promise as the two ongoing repositories of software projects in the SE community, both lacking structured documentation, which hinders researchers to identify the datasets that are suitable for their purposes. Déry and Abran [22] mention that "a key challenge in data analysis using the ISBSG repository (...) is to assess the consistency of the effort data collected".

5.2 Effort versus Cost

Effort and cost are in many studies used as equivalent; usually cost is mentioned, where actually effort is meant. An example is the definition by Buglione and Ebert [36]: "An estimate is a quantitative assessment of a future endeavor's likely cost or outcome." [36]. Petersen argues that both are an important decision criterion for companies [2]. A recent study on productivity in agile software development [37] mentions cost as one of nine highly-related productivity dimensions, although the original study by Melo et al. [38] of two agile teams revealed that most team members did not share the same understanding of productivity. As a consequence of the blurred distinction between effort and cost, the latter is for a major part missing in studies on project performance. Radliński [3] gives an overview of variables used in analysis within the ISBSG repository, yet cost is not mentioned. When cost actually is mentioned, it often is as equivalent to or a derivative of effort, see for example [5], [6], and [4]. Deng and MacDonell [39] propose an approach based on justified normalization of functional size, to challenge questions among researchers about the quality and completeness in the ISBSG-repository.

Both the EBSPM-repository and the ISBSG-subset use functional size (Function Points) as metric to normalize the projects in their repositories. Our study shows that in both repositories Size is strongly significant for Cost of software projects, an effect well known from other related work [40]. This emphasizes the importance of including functional size in any form in a software project repository. It might even be an important next step to automate the counting of functional size, although recent research indicated that stakeholders on functional size measurement do not see a direct need for this [41]. However, we assume that this opinion might be slightly tainted by self-interest.

Although we used regression analysis for our study a warning is in place: Jørgensen and Kitchenham [42] argue that violations of essential regression model assumptions in research studies to a large extent may explain disagreement among researchers on economy of scale effects or diseconomy of scale effects with regard to size and effort. Randomized controlled experiments with fixed software sizes and random allocation of development of software of different sizes, and the use of more in-depth of analyses of software projects might help here [42]. Radliński [3] examined how various project factors in ISBSG are related with the number of defects, finding that there are very few factors significantly influencing this aspect of software quality. Such results might suggest that software quality depends rather on a wider set of factors [3]. This confirms what we found with regard to number of defects in the ISBSG-subset. Finally, no scientific studies are published that examine the link between effort and cost of software projects based on real industry data.

6 CONCLUSIONS

We compared two industrial yet publicly available software project repositories, the EBSPM-repository and a subset of the ISBSG-repository, in order to analyze differences with regard to Cost, Size, Duration, and Number of Defects. We determined suitability of some key variables (Size, Duration and Number of Defects) of both data sets for the purpose of cost prediction.

We identified three key findings: (1) We demonstrate a (log)-linear relation between cost on the one hand, and size, duration and number of defects on the other. This justifies conducting linear regression for cost. (2) We establish that ISBSG is substantially different from, e.g., EBSPM, in terms of cost (cheaper) and duration (faster), and the relation between cost and effort. This implies that practitioners and researchers alike should be cautious when drawing conclusions from a single repository. (3) We show that while in ISBSG effort is the most important cost factor, this is not the case in other repositories, such as EBSPM in which size is the dominant factor.

We showed that effort and cost of software projects in the ISBSG-subset are interrelated, although results might be influenced by definitional issues with regard to cost and because we examined a subset of only 3% of the ISBSG-repository. We argue that, supported by the importance of both effort and cost data for decision makers in industry, effort and cost should be treated as different metrics in research.

ACKNOWLEDGMENTS

Our thanks to Tableau for allowing us to use their BI solution to build the EBSPM-tool and ISBSG for allowing us to use their repository for research purposes.

REFERENCES

[1] D. Šmite, F. Calefato en C. Wohlin, „Cost-Savings in Global Software Engineering: Where's the Evidence," *IEEE Software,* vol. 32, nr. 4, pp. 26-32, 2015.

[2] K. Petersen, „Measuring and predicting software productivity: A systematic map and review," *Information and Software Technology,* vol. 53, nr. 4, pp. 317-343, 2011.

[3] Ł. Radliński, „Factors of Software Quality–Analysis of Extended ISBSG Dataset," *Foundations of Computing and Decision Studies,* vol. 36, nr. 3-4, pp. 293-313, 2011.

[4] R. Jeffery, M. Ruhe en I. Wieczore, „A comparative study of two software development cost modeling techniques using multi-organizational and company-specific data," *Information and software technology,* vol. 42, nr. 14, pp. 1009-1016, 2000.

[5] P. C. Pendharkar en J. A. Rodger, „The relationship between software development team size and software development cost," *Communications of the ACM,* vol. 52, nr. 1, pp. 141-144, 2009.

[6] B. Czarnacka-Chrobot, „The role of benchmarking data in the software development and enhancement projects effort planning," in *Proceedings of the 2009 conference on New Trends in Software Methodologies, Tools and Techniques,* 2009.

[7] C. Jones, "Sources of Software Benchmarks," Capers Jones & Associates, 2011.

[8] T. Menzies en T. Zimmermann, „Software Analytics: So What?," *IEEE Software*, vol. July/August, pp. 31-37, 2013.

[9] A. Bala en A. Abran, „Use of the Multiple Imputation Strategy to Deal with Missing Data in the ISBSG Repository," *Journal of Information Technology & Software Engineering*, vol. 6, nr. 1, pp. 1-12, 2016.

[10] ISBSG, „International Software Benchmarking Standards Group," 2014. [Online]. Available: http://www.isbsg.org/.

[11] M. Jorgensen en M. Shepperd, „A Systematic Review of Software Development Cost Estimation Studies," *IEEE Transactions on Software Engineering*, vol. 33, nr. 1, pp. 33-53, 2007.

[12] H. Huijgens, G. Gousios en A. v. Deursen, „Pricing via Functional Size - A Case Study of a Company's Portfolio of 77 Outsourced Projects," in *IEEE 9th International Symposium on Empirical Software Engineering and Measurement (ESEM 2015)*, 2015.

[13] H. Huijgens, „Evidence-Based Software Portfolio Management: A Tool Description and Evaluation," in *ACM Proceedings of the 20th International Conference on Evaluation and Assessment in Software Engineering*, Limerick, Ireland, 2016.

[14] H. Huijgens, R. v. Solingen en A. v. Deursen, „How to build a good practice software project portfolio?," in *ACM Companion Proceedings of the 36th International Conference on Software Engineering (ICSE)*, 2014.

[15] H. Huijgens, A. v. Deursen en R. v. Solingen, „An Exploratory Study on the Effects of Perceived Value and Stakeholder Satisfaction on Software Projects," in *IEEE/ACM Proceedings of the International Conference on Evaluation and Assessment in Software Engineering (EASE)*, 2016.

[16] H. Huijgens, L. Minku, C. Lokan en A. v. Deursen, „Cost of Software Engineering Projects: A Comparison of two Software Project Repositories - Technical Report TUD-SERG-2016-017," Software Engineering Research Group (SERG), Delft University of Technology, 2016.

[17] R. Nisbet, G. Miner en J. Elder IV, Handbook of Statistical Analysis and Data Mining Applications, Academic Press, 2009.

[18] ISBSG, C. Jones en Reifer Consultants, „The Impact of Software Size on Productivity," ISBSG.

[19] H. Huijgens en F. Vogelezang, „Do Estimators Learn? On the Effect of a Positively Skewed Distribution of Effort Data on Software Project Productivity," in *ACM Proceedings of the 7th International Workshop on Emerging Trends in Software Metrics (WETSoM)*, Austin, Texas, USA, 2016.

[20] T. Foss, E. Stensrud, B. Kitchenham en I. Myrtveit, „A Simulation Study of the Model Evaluation Criterion MMRE," *IEEE Transactions on Software Engineering*, vol. 29, nr. 11, pp. 985-995, 2003.

[21] K. A. Bollen en R. W. Jackman, „Regression Diagnostics: An Expository Treatment of Outliers and Influential Cases," in *Modern Methods of Data Analysis*, ISBN 0-8039-3366-5 red., Newbury Park, CA, Sage, 1990, p. 257–91.

[22] D. Déry en A. Abran, „Investigation of the effort data consistency in the ISBSG repository," in *Proceedings of the 15th Intern. Workshop on Software Measurement*, 2005.

[23] R. Jeffery, M. Ruhe en I. Wieczore, „Using public domain metrics to estimate software development effort," in *IEEE Seventh International Software Metrics Symposium (METRICS)*, 2001.

[24] L. C. Briand, T. Langley en I. Wieczorek, „A replicated assessment and comparison of common software cost modeling techniques," in *ACM Proceedings of the 22nd International Conference on Software Engineering*, 2000.

[25] I. Wieczorek en M. Ruhe, „How valuable is company-specific data compared to multi-company data for software cost estimation?," in *IEEE Proceedings of the Eighth IEEE Symposium on Software Metrics*, 2002.

[26] C. Lokan en E. Mendes, „Cross-company and single-company effort models using the ISBSG database: a further replicated study," in *Proceedings of the 2006 ACM/IEEE international symposium on Empirical software engineering*, 2006.

[27] L. Minku, E. Mendes en F. Ferrucci, „How to make best Use of Cross-Company Data for Web Effort Estimation?," in *ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)* , 2015.

[28] E. Mendes, C. Lokan, R. Harrison en C. Triggs, „A replicated comparison of cross-company and within-company effort estimation models using the ISBSG database," in *IEEE 11th IEEE International Symposium on Software Metrics*, 2005.

[29] M. Garre, J. Cuadrado, M. Sicilia, M. Charro en D. Rodríguez, „Segmented parametric software estimation models: Using the em algorithm with the isbsg 8 database," *Information Technology Interfaces*, 2005.

[30] L. Minku, „On the Terms Within- and Cross-Company in Software Effort Estimation," in *ACM Proceedings of the The 12th International Conference on Predictive Models and Data Analytics in Software Engineering*, 2016.

[31] A. Abran, R. Dumke, J. Desharnais, I. Ndyaje en C. Kolbe, „A strategy for a credible & auditable estimation process using the ISBSG International Data Repository," in *IWSM*, 2002.

[32] C. Lokan, T. Wright, P. Hill en M. Stringer, „Organizational benchmarking using the ISBSG data repository," *IEEE Software*, vol. 18, nr. 5, pp. 26-32, 2001.

[33] M. Fernández-Diego en F. González-Ladrón-de-Guevara, „Potential and limitations of the ISBSG dataset in enhancing software engineering research: A mapping review," *Information and Software Technology*, vol. 56, nr. 6, pp. 527-544, 2014.

[34] S. Oligny, P. Bourque, A. Abran en B. Fournier, „Exploring the relation between effort and duration in software engineering projects," in *Proceedings of the World Computer Congress*, 2000.

[35] L. Cheikhi en A. Abran, „Promise and ISBSG Software Engineering Data Repositories: A Survey," in *Joint Conference of the International Workshop on Software Measurement and the 2013 Eighth International Conference on Software Process and Product Measurement (IWSM-MENSURA)*, Ankara, Turkey, 2013.

[36] L. Buglione en C. Ebert, „Estimation tools and techniques," *IEEE Software*, vol. 28, nr. 3, pp. 91-94, 2011.

[37] S. M. A. Shah , E. Papatheocharous en J. Nyfjord, „Measuring productivity in agile software development process: a scoping study," in *ACM Proceedings of the 2015 International Conference on Software and System Process (ICSSP)*, 2015.

[38] C. Melo, D. Cruzes, F. Kon en R. Conradi, „Agile team perceptions of productivity factors," in *IEEE Agile Conference (AGILE)*, 2011.

[39] K. Deng en S. G. MacDonell, „Maximising data retention from the ISBSG repository," in *Proceedings of the twelfth International Conference on Evaluation and Assesment of Software Engineering (EASE)*, 2008.

[40] C. Gencel en O. Demirors, „Functional Size Measurement Revisited," *ACM Transactions on Software Engineering and Methodology*, vol. 17, nr. 3, pp. 15:1-15:36, June 2008.

[41] H. Huijgens, M. Bruntink, A. v. Deursen, T. v. d. Storm en F. Vogelezang, „An Exploratory Study on Automated Derivation of Functional Size based on Code," in *ACM Proceedings of the International Conference on Software and Systems Process (ICSSP)*, Austin, Texas, USA, 2015.

[42] M. Jørgensen en B. Kitchenham, „Interpretation problems related to the use of regression models to decide on economy of scale in software development," *Journal of Systems and Software*, vol. 85, nr. 11, pp. 2494-2503, 2012.

[43] L. Minku en X. Yao, „Ensembles and Locality: Insight on Improving Software Effort Estimation," *Information and Software Technology, Special Issue on Best Papers from PROMISE 2011*, vol. 55, nr. 5, pp. 1512-1528, 2013.

[44] B. Turhan en E. Mendes, „A comparison of cross- versus single- company effort prediction models for web projects," in *Euromicro Conference on Software Engineering and Advanced Applications*, Verona, Italy, 2014.

[45] E. Kocaguneli, T. Menzies en E. Mendes, „Transfer learning in effort estimation," *Empirical Software Engineering*, vol. 20, nr. 3, pp. 813-843, 2015.