ISSN 0361-7688, Programming and Computer Software, 2024, Vol. 50, No. 8, pp. 786–795. © Pleiades Publishing, Ltd., 2024.

Merging Distinct Sources Databases to Improve Software Estimation Models

Francisco Valdés-Souto[a,*] (ORCID: 0000-0001-6736-0666) and Jorge Valeriano-Assem[b,]**

[a] *National Autonomous University of Mexico Science Faculty, CDMX, México*

[b] *SPINGERE, CDMX, México*

**e-mail: fvaldes@ciencias.unam.mx*

***e-mail: jorge.valeriano@spingere.com.mx*

Received May 8, 2024; revised August 17, 2024; accepted September 12, 2024

Abstract—Context. For more than six decades, software cost/effort estimation has been a relevant topic for research due to its impact on the industry. Although many estimation models exist, regression-based estimation approaches have been predominantly used in the literature. However, some problems have been observed both in industry and academia: the lack of datasets with a high or at least enough number of data points and the arbitrary combination of different source databases belonging to practitioners in order to create larger datasets.

Objective. Propose the application of the Kruskal–Wallis test to validate the integration of distinct source databases (independent groups), thereby avoiding the mixing of unrelated data, increasing the number of data points, and improving the estimation models.

Method. We conducted a case study using real data from an international company, specifically data from their Mexico office. This office provides software development services for a technological tower identified as "Microservices and APIs." The data were collected in 2020.

Results: The quality criteria in the final estimation model were improved. The MMRE was reduced by 25.4% (from 78.6 to 53.2%), the standard deviation was reduced by 97.2% (from 149.7 to 52.5%), and the Pred (25%) indicator increased by 3.2 percentage points. Additionally, the number of data points increased significantly, and linear regression constraints was accomplished. The application of the Kruskal–Wallis test to validate the integration of distinct source databases (independent groups) proved useful in improving the estimation models.

Keywords: linear regression model, software estimation, effort estimation, cost estimation, functional size, COSMIC method, Kruskal–Wallis

DOI: 10.1134/S0361768824700762

1. INTRODUCTION

The importance of software cost/effort estimation in several areas, such as planning, budgeting, control, and software project success, has made it a pertinent topic for industry and study throughout the past few decades. Estimation performance comparison is one of the primary subjects in the literature [1], with regression-based estimation approaches being the most frequently utilized in research and developed with different databases [1, 2]. One of the main concerns in the literature [3] is the small quantity of data points in the datasets. However, small datasets are more common than expected in the industry. Another issue observed in the industry is that different source databases belonging to researchers or practitioners are often arbitrarily combined without any evaluation to guarantee their utility.

Estimation, a topic extensively explored in the field of statistics, holds immense potential for enriching, aiding, and expediting the advancement of software engineering. By leveraging current knowledge, we can pave the way for innovative solutions and improved practices.

This work specifically proposes a method for using widely established statistical methods to validate the integration of distinct source databases. The aim is to improve estimation models by increasing the number of data points through database combination.

In the literature reviewed, does not find any approach to handling this issue formally for software estimation.

The outline of this paper is as follows. Section 2 provides background information related to software estimation and database problems and considers the statistical elements used in the paper. Section 3 pro-

poses a procedure used in a case study to evaluate data source integration, including statistical validation, and demonstrating the improvements generated in the estimation model. In Section 4, the conclusions are discussed.

2. BACKGROUND

2.1. Parametric Software Estimation

Software estimation has existed for more than 70 years, first appearing in the 1950s [4]. Several researchers have looked into it and discovered, among other things, that it "is central to the success of a development project" [5], "is one of the most serious problems that cause the software projects failure" [4], "is one of the most crucial activities of software development" [6], and "has a crucial impact on budgeting and project planning in the industry" [3]. The literature on software estimation encompasses a wide range of techniques developed over the course of more than six decades, resulting in a variety of estimation methods [7, 8], numerous classifications of estimation methods [1, 3, 5, 9–11], and topologies for estimation processes [12, 13].

Software estimation research has received considerable attention and is important to the industry. Despite this, many unanswered questions and challenges with estimation remain [3, 14]. As mentioned by [4], several authors identify the measurement of software size as a significant factor in the accuracy of estimates [15–18]. Nowadays, functional size [19] is the only software feature that can be agreed upon and, thus, quantified consistently, which strengthens this position.

Every estimating model closely connects to the method used to measure the input variables used to produce the estimate [9].

2.1.1. Database conformation for parametric estimation. When creating an estimation model, it is necessary to integrate a reference database based on previously completed projects. This database contains project data that enables the identification of correlations between several variables (cost drivers), with functional size being the primary variable. Several authors have mentioned problems encountered while creating regression-based models [4, 9, 11, 20–22].

Replication of findings can often be difficult, even though most of the examined literature uses regression-based estimating techniques based on reference databases [1, 2, 23]. It is generally accepted that projects to be estimated should be represented by previous projects that adhere to a reference database [1]. According to Kitchenham et al. [24], no cost estimating methodology–or any other model, for that matter–will forecast well if asked to estimate effort for projects that are significantly different from the projects on which the model was developed. Researchers "have used databases documented based on the past completed projects they participated in," according to Valdés [25]. Typically, not everyone has access to this information; it can be challenging to obtain, or some of its components may not make sense to all database users considering they are significantly different from the projects. Several authors have identified weaknesses in the datasets from which the estimation models were created [1, 3, 22, 24].

One of the fundamental and complex problems we face is the lack of datasets with a high (enough) number of data points, a fundamental statistical principle. This was meticulously analyzed by Carbonera et al. [3], who classified the number of data points in datasets as high quality (more than 15 points), medium quality (10 to 15 data points), or low quality (less than 10 data points), highlighting the challenge of dataset quality we often encounter.

According to the central limit theorem, under very general conditions, if Sn is the sum of n independent random variables with finite mean and variance, then the distribution function of Sn "approximates well" a normal distribution. Empirically, at least 30 data points are considered for each independent variable [26].

Collecting 30 similar projects in the industry according to specific features is a challenge. Morgenshtern et al. [27] mention, "Algorithmic models need historical data, and many organizations do not have this information. Additionally, collecting such data may be both expensive and time-consuming." Another problem observed in the industry is that different source databases belong to practitioners or researchers, yet they are arbitrarily combined without any evaluation to guarantee their utility.

Additionally, there are organizations with datasets, such as the International Software Benchmarking Standards Group (ISBSG, www.isbsg.org), a nonprofit organization that collects data about projects and centralizes an international database used as a reference. Also, the Mexican Software Metrics Association (AMMS, www.amms.org.mx) has a reference database integrated through an online survey platform with real Mexican industry projects. The database information is related to software projects in Mexico (already concluded). Given this, it is relevant to define a technique or mechanism that appropriately uses statistical methods to allow the integration of distinct databases, aiming to solve the issue of a few data points in databases to generate estimation models.

A description of the AMMS dataset is defined in [22], the features included in the AMMS dataset are similar to the features collected in the ISBSG dataset, the ISBSG dataset information could be obtained in www.isbsg.org.

2.1.2. Estimation models performance comparison. The literature compares estimation models' performances by focusing on the application of quality criteria to provide confidence in a particular estimation model. When the estimation models are applied mul-

tiple times, the discrepancy between the estimated and actual values is recorded and evaluated using defined quality criteria to assess the models' confidence level.

The most commonly used criteria in the software estimation literature are:

- Mean magnitude of relative error (MMRE),
- Standard deviation of MRE (SDMRE),
- Prediction level, PRED ($x\%$),
- Median magnitude of relative error (MdMRE),
- Mean absolute residual (MAR) [28].

Some researchers have analyzed these techniques and have identified some concerns about them [24, 29–34].

2.2. Kruskal–Wallis Test

The Kruskal–Wallis test is a nonparametric statistical method used to compare the distributions of independent groups [35]. It is an alternative to the parametric one-way analysis of variance (ANOVA) when the assumptions of normality and homogeneity of variances are violated or when the data are measured on an ordinal scale. Named after William Kruskal and Wilson Wallis, who introduced it in 1952 [36], this test is particularly valuable when the data do not meet the assumptions ANOVA requires.

The Kruskal–Wallis test operates by rank-ordering the combined data from all groups, converting the original data values to ranks. It then computes a test statistic H based on the ranks, where a higher H value indicates greater evidence against the null hypothesis of no difference among group distributions. Under the null hypothesis, H follows a chi-square distribution with $k - 1$ degree of freedom, where k is the number of groups being compared [35].

This test essentially evaluates whether the distributions of distinct sample ranks differ significantly, suggesting differences in the population medians. Suppose the calculated H value exceeds the critical value from the chi-square distribution. In that case, it indicates that at least one group differs significantly from the others in terms of location, prompting rejection of the null hypothesis.

The Kruskal-Wallis is commonly used in experimental studies to compare outcomes across multiple treatment groups or in observational studies to compare outcomes across different populations. It is a robust method for comparing multiple groups, particularly when the data violate the assumptions of parametric tests or when the data are ordinal in nature.

2.3. Outliers Identification Using Tukey Test

The Tukey test, also known as Tukey's range test, is a statistical method commonly used for identifying outliers in a dataset. Developed by John Tukey [37], this test provides a systematic way to detect observations that deviate significantly from the rest of the data.

The procedure involves calculating the interquartile range (IQR), which is the difference between the third quartile (Q3) and the first quartile (Q1) of the data distribution. Then, a threshold is established based on the IQR, typically defined as 1.5 times the IQR. Observations that fall below Q1 minus the threshold or above Q3 plus the threshold are flagged as potential outliers.

The Tukey test is robust against moderate departures from normality and is particularly useful for identifying outliers in skewed or non-normally distributed datasets. It provides a reliable and straightforward approach to detecting outliers, helping to ensure the integrity and validity of statistical analyses [38].

3. CASE STUDY: MERGIN DISTINCT SOURCES DATABASES

This section provides a general overview of the case study conducted at an international company with a Mexico office (referred to as COMPANY for confidentiality reasons). This office have offer software development services in a financial institution The data was collected in 2020.

3.1. Case Study Steps

The case study consists of five steps; however, the paper will describe only the last two:

1. **Projects identification/ classification**. The COMPANY performed this step to define the type of projects they needed to estimate. They selected projects from a technological tower identified as "Microservices and APIs."

2. **Projects information collection**. The company identified information about past projects, including the functional user requirements (FUR) and the effort expended on them. However, they collected information for only eight (8) projects.

3. **Functional size approximation**. With the information integrated, we use the EPCU approximation approach [39] to measure the FURs using COSMIC (ISO/IEC 19761) for each project.

4. **Integration of additional projects from other sources**. Since the projects provided by the COMPANY were not enough to build a robust estimation model, we searched for similar projects as provided by the COMPANY, that means, related to microservices or API development in the ISBSG database and the AMMS databases. Forty-nine (49) projects exist in the databases: 15 from the ISBSG database and 34 from the AMMS database. See **Annex 1**.

5. **Building the final estimation model**. To build the final estimation model, we used the Kruskal–Wallis test to compare the distributions of independent groups to evaluate whether the integration was feasi-

Table 1. (a) Sample size by source, "Microservices and APIs" projects. (b) Total functional size by source

SOURCE	Sample Size	%
COMPANY	8	14.0%
ISBSG	15	26.3%
AMMS	34	59.7%
Total	57	100.0%

a)

SOURCE	COSMIC Functional Size (CFP)	%
COMPANY	2418.7	11.0%
ISBSG	3873	17.6%
AMMS	15674.6	71.4%
TOTAL	21966.3	100.0%

b)

ble. Then, we followed the steps proposed by Valdés-Souto et al. in [25, 22] to build and improve the estimation model.

3.2. Projects Characterization

The COMPANY and alternative sources (ISBSG, AMMS) play a pivotal role in our project characterization. They provide the necessary information to select similar projects in its features that enable us to compare the size and effort required. The effort was gathering using COSMIC (ISO/IEC 19761) and is the base metric of our project characterization.

Table 1.a shows in column 1 the acronym of the source to which the projects were obtained, in column 2 the number of projects in the sample, and in column 3 the proportion of each group considering the total number of projects. All the projects were related to microservices or APIs development.

Table 1.b shows in column 1 the acronym of the source to which the projects belong, in column 2 the functional size in CFP per group, and in column 3 the proportion of the size per group concerning the total functional size in the sample.

From the tables above, it is possible to observe that the AMMS database contributes significantly, accounting for 71.4% of the total functional size and 59.7% of the total projects. The second major contributor in quantity of projects and size is the ISBSG database with 26.3% of the total projects and 17.4% of the total size.

The data from the COMPANY was the minimum contribution in size and quantity. Because the central limit theorem, the number of projects it is not enough to build a significant estimation model using only the data provided initially by the COMPANY.

Originally, the estimation model that could be built with only the COMPANY's data is shown in Fig. 1, where the "x" axis corresponds to CFP and the "y" axis corresponds to effort.

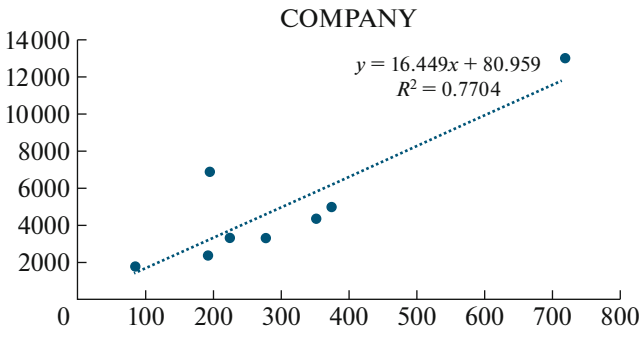

Fig. 1. COMPANY estimation model.

Although the model get a R2 over 77%, the number of data do not allow to extrapolate the conclusions.

Usually, the natural and arbitrary step is to build an estimation model using the three datasets; the model with three datasets is shown in Fig. 2.

In this case the model presents a R2 of 62% that it is lower than 77% of the initial model. The main concern is if the added dataset has a distinct distribution or the mean has significant variation of the mean of pre-

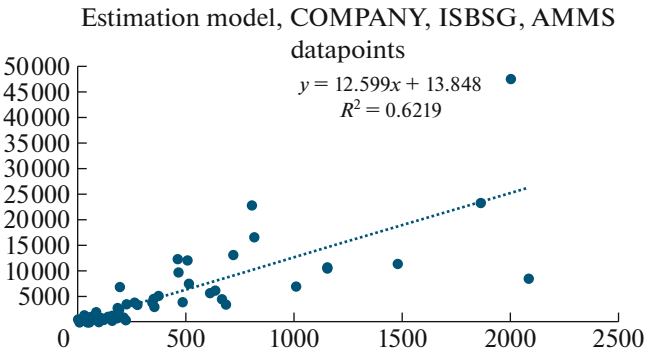

Fig. 2. COMPANY, ISBSG, AMMS estimation model.

Table 2. Kruskal–Wallis test results for three datasets

N	57
Degrees of freedom (Number of sets –1)	2
Asymp.sig. (p-value)	0.00001696

vious data, the impact could be relevant and maybe is not good idea to integrate the data.

3.3. Feasibility of Integration

The next step was to compare the distributions of independent groups to evaluate whether the integration was feasible with solid statistical foundations. Specifically, we assessed whether the distributions of the three databases (COMPANY, ISBSG, AMMS) at the PDR variable (HH/CFP) are the same or different. This evaluation allows us to determine whether it is appropriate to combine the three databases into a single database and build estimation models. Since the project samples come from different databases, they are known in statistics as independent samples. In this case, there are three samples. To assess these, we used a nonparametric test called the Kruskal–Wallis test [35, 36], which allows us to conclude whether the distributions of the three samples are equal or different.

The null hypothesis (**H0**) is that no significant difference exists between the COMPANY, ISBSG, and AMMS database distributions. The alternative hypothesis (**H1**) is: At least one distribution of the COMPANY, ISBSG, or AMMS databases is significantly different.

The significance level required is at least $\alpha = 0.05$. If the test's p-value is greater than or equal to 0.05, the hypothesis **H0** is correct; otherwise, if it is less than 0.05, the alternative hypothesis **H1** is correct. The Kruskal-Wallis test was executed using the statistical software SPSS® version 25 in Spanish.

Table 2 summarizes the results of the Kruskal-Wallis test for the COMPANY, ISBSG, and AMMS databases. The p-value is less than 0.05 (LINE 3); therefore, the null hypothesis (**H0**) is rejected. Consequently, we conclude that there is a significant difference in at least one of the distributions of the COMPANY, ISBSG, and AMMS databases, as stated by the alternative hypothesis (**H1**).

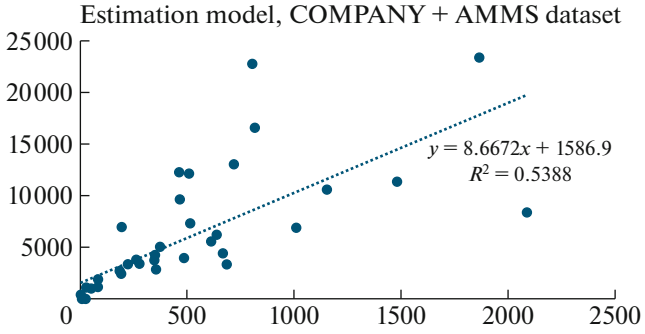

Fig. 3. Initial estimation model AMMS-COMPANY dataset.

Table 3. Kruskal–Wallis test results by couple of datasets

Pair	Asymp.sig. (p-value)	Asymp.sig. (p-value) with Bonferroni correction.
ISBSG – AMMS	0.00001578	0.00004735
ISBSG – COMPANY	0.0006197	0.001859
AMMS – COMPANY	0.2057	0.6171

Consequently, to determine which databases have different distributions, it is necessary to perform pairwise comparisons using the Kruskal–Wallis test, adjusting the resulting *p*-value to account for the number of tests. This correction is known as the Bonferroni correction [40]. Table 3 shows the results for each pair of datasets analyzed. The AMMS–COMPANY pair is the only one with an adjusted *p*-value (0.6171) greater than 0.01667 (0.05/3). From this, we conclude that the AMMS and COMPANY databases have the same distribution, while the ISBSG database has a different distribution. Consequently, it is only possible to integrate the COMPANY and AMMS datasets to build the estimation model with 42 datapoints (COMPANY (8), AMMS (34)).

3.4. Building the Estimation Model

Once the integration validation is performed, we have the final dataset (COMPANY + AMMS) to develop an estimation model directly. The results are shown in Fig. 3.

The generated estimation model is $y = 8.6672x + 1586.9$, with a determination coefficient $R^2 = 0.5388$.

For comparison purposes, the estimation model using only the COMPANY data points is $y = 16.449x + 80.959$, with a determination coefficient $R^2 = 0.7704$, but the number of data points is not sufficient.

However, it is needed to develop a linear regression model validation and diagnostics [22].

The normal probability graph and the residuals graph were obtained using Excel complement to analyze the regression model.

Figure 4 shows evidence against normality, as the points do not follow the identity line in the normal probability graph. Additionally, in the residuals graph, the variance of the residuals increases as the fitted values increase, showing a systematic pattern and indicating non-constant variance, which means the data do not exhibit homoscedasticity. In conclusion, the

Table 4. Quality criteria comparison for estimation models

	$y = 16.449x + 80.959$ $R^2 = 0.7704$	$y = 8.6672x + 1586.9$ $R^2 = 0.5388$	$y = x^{0.9422} \cdot e^{3.0156}$
N	8	42	38
MMRE	27.8%	78.6%	53.2%
SDMRE	13.4%	149.7%	52.5%
Pred(25%)	50.0%	31.0%	34.2%
Enough data	NO	YES	YES

regression model is not appropriate for this dataset and needs improvement.

Looking for transformation to correct the assumptions of the model, we used the logarithm for the functional size, effort variables and build a new estimation model. See Fig. 5, where the x axis corresponds to Log(CFP) and the y axis correspond to Log(effort). The estimation model generated is Log(y) = 0.9326 Log(x) + 2.8916, with a determination coefficient $R^2 = 0.8339$.

We conducted validation and diagnostics using the transformed data, with the results shown in Fig. 6. The plot of fitted values against the residuals shows constant variance, as the dots do not display patterns, indicating homoscedasticity. Additionally, most dots follow the identity line pattern, representing normality. Consequently, the estimation model shown in Fig. 5 is useful because the dataset have enough data points and the statistical principles of normality and homoscedasticity are accomplished, then utilizing a linear regression model with this data set is correct.

After that, we search for outliers using the Tukey test, finding four (4) outliers as shown in Fig. 7.

After removing the outliers, a new estimation model was obtained, the result is:

Log(y) = 0.9377 Log(x) + 2.8996 with a Determination coefficient $R^2 = 0.9023$. The model uses logarithmic variables. To apply it to the actual variables, we need to eliminate the logarithmic transformation using the inverse operation (Euler's number, e), resulting in the final model: $y = x^{0.9377} \cdot e^{2.8996}$.

Table 4 presents the quality criteria for the developed estimation models. The best model is the last one, achieved after applying validations and diagnostics, then performing a transformation and removing the outliers.

All the quality criteria were improved, and the model triumphantly satisfied the conditions of having enough data points and the statistical principles of normality and homoscedasticity to use linear regression.

4. ANALISYS

In the case study presented, the COMPANY under study had only eight (8) data points. Two additional datasets were considered: the ISBSG dataset with fifteen (15) data points and the AMMS dataset with thirty-four (34) data points.

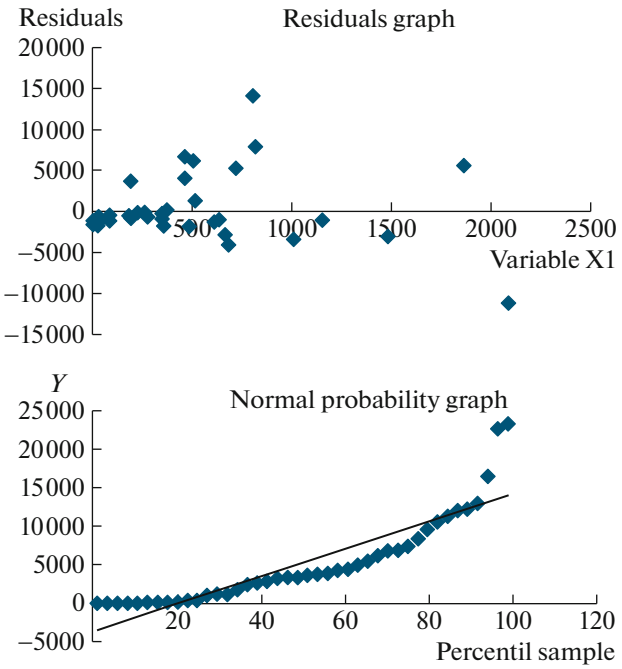

Fig. 4. Graph for validation and diagnostics AMMS-COMPANY dataset.

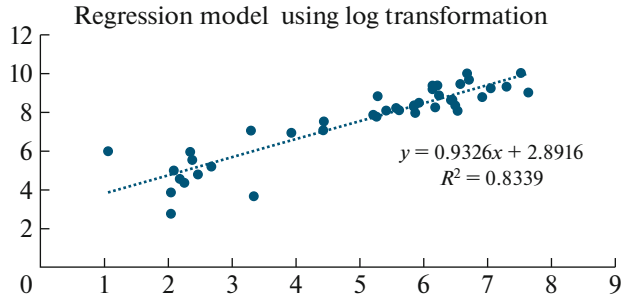

Fig. 5. Estimation model AMMS-COMPANY dataset using logarithm transformation.

Table 5

Project Number	BASE	ID	Functional Size COSMIC (CFP)	Total Effort (HH)	PDR [HH/CFP]	Project Number	BASE	ID	Functional Size COSMIC (CFP)	Total Effort (HH)	PDR [HH/CFP]
1	COMPANY	1	374.2	5040	13.47	31	AMMS	8	806.2	22743	28.21
2	COMPANY	2	193.1	2440	12.64	32	AMMS	9	10.4	394	37.88
3	COMPANY	3	195	6928	35.53	33	AMMS	10	26.9	1193.8	44.38
4	COMPANY	4	352.3	4368	12.40	34	AMMS	11	10.8	260	24.07
5	COMPANY	5	718	13048	18.17	35	AMMS	12	50.8	1034	20.35
6	COMPANY	6	277	3400	12.27	36	AMMS	13	8.1	148	18.27
7	COMPANY	7	84.6	1848	21.84	37	AMMS	14	7.7	16	2.08
8	COMPANY	8	224.5	3360	14.97	38	AMMS	15	28.4	40	1.41
9	ISBSG	1	46	20	0.43	39	AMMS	16	7.7	48	6.23
10	ISBSG	2	114	429	3.76	40	AMMS	17	665.3	4414	6.63
11	ISBSG	3	97	20	0.21	41	AMMS	18	2085.5	8400	4.03
12	ISBSG	4	177	694	3.92	42	AMMS	19	9.5	79	8.32
13	ISBSG	5	80	355	4.44	43	AMMS	20	1479.9	11354	7.67
14	ISBSG	6	118	489	4.14	44	AMMS	21	636.8	6182	9.71
15	ISBSG	7	57	20	0.35	45	AMMS	22	1010.6	6882	6.81
16	ISBSG	8	155	491	3.17	46	AMMS	23	84	1192	14.19
17	ISBSG	9	135	600	4.44	47	AMMS	24	684.6	3416	4.99
18	ISBSG	10	209	606	2.90	48	AMMS	25	345.9	3733	10.79
19	ISBSG	11	223	509	2.28	49	AMMS	26	183.6	2682	14.61
20	ISBSG	12	157	899	5.73	50	AMMS	27	8.8	96	10.91
21	ISBSG	13	2003	47493	23.71	51	AMMS	28	2.9	408	140.69
22	ISBSG	14	142	669	4.71	52	AMMS	29	14.6	184	12.60
23	ISBSG	15	160	784	4.90	53	AMMS	30	11.8	120	10.17
24	AMMS	1	464.8	9656.4	20.78	54	AMMS	31	611.9	5568	9.10
25	AMMS	2	1154.1	10563	9.15	55	AMMS	32	484	3919.1	8.10
26	AMMS	3	1864.1	23358	12.53	56	AMMS	33	353.7	2865	8.10
27	AMMS	4	507.5	12111	23.86	57	AMMS	34	262	3776	14.41
28	AMMS	5	463.2	12264	26.48						
29	AMMS	6	513.2	7373	14.37						
30	AMMS	7	815.7	16566	20.31						

However, the ISBSG data points were rejected by the Kruskal–Wallis test, resulting in a final dataset with forty-two (42) data points. After removing outliers, the dataset contained thirty-eight (38) data points.

The results obtained are related to MMRE that was reduced by 25.4% (from 78.6 to 53.2%), the standard deviation was reduced by 97.2% (from 149.7 to 52.5%), and the Pred(25%) indicator increased by 3.2 percentage points. Notably, the number of data points was significantly increased, from 8 to 38 (475%), bolstering the robustness of our findings.

5. CONCLUSIONS

Because of its influence on the industry, software cost/effort estimate has been a pertinent research issue for over 60 years, with regression-based estimation methodologies being the most widely used. Nonetheless, several issues have been noted in academia and industry, particularly with regard to conforming datasets:

- The lack of datasets with a high (enough) number of data points (a fundamental statistical principle)
- Different source databases belonging to practitioners are often arbitrarily combined to create a larger dataset.

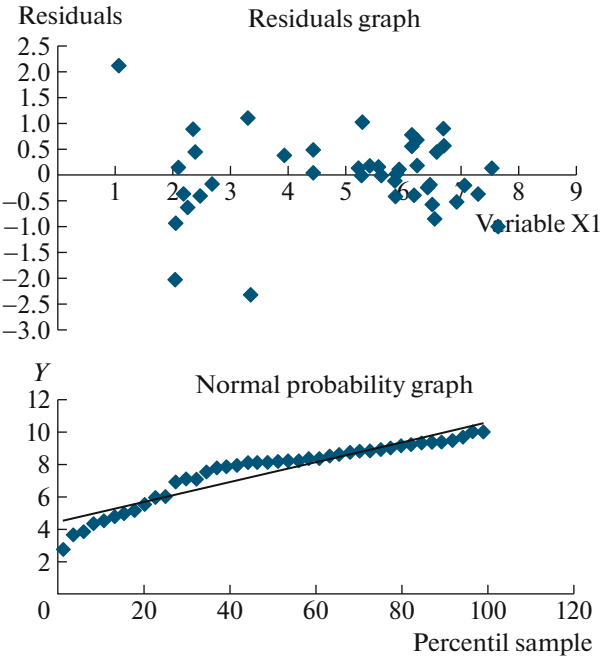

Fig. 6. Graph for validation and diagnostics AMMS-COMPANY dataset with logarithmic transformation.

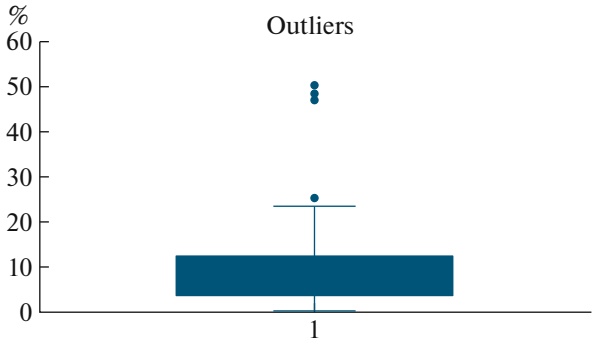

Fig. 7. Outliers AMMS-COMPANY dataset with logarithmic transformation

This work presents a real case study, applying the necessary statistical formalisms to decide whether or not to integrate information from distinct sources databases (independent groups) using the Kruskal-Wallis test to validate the integration, aiming to merge those without differences among group distributions and avoid mixing incompatible data.

This proposal allows the integration of distinct data sources through validation analysis, increasing the number of data points to obtain a significant dataset and improve the generated estimation models.

As demonstrated by the case study with real industry datasets, the generated estimation model was superior to the original (using three datasets without analysis), confirming that integration without validation and diagnostics is suboptimal. It is essential to mention that the scope of this work was to establish the foundations of a formal methodology for generating reliable and useful estimation models based on distinct data sources, addressing a common problem in organizational practice. However, a lack of correct statistical principles, for example if regression techniques are not applied correctly, it is possible to generate also valueless estimation models. The proposed approach was used in conjunction with statistical principles in the case study presented and developed previously [22].

Finally, this work addresses a frequent problem in industry and academia and establishes the foundations of a novel formal methodology imported from the statistics area and applied in the software estimation discipline, the literature reviewed does not find any approach applied to software estimation to solve the small size of datasets. This methodology is designed to generate reliable and useful estimation models from distinct data sources, marking a significant contribution to the field.

APPENDIX 1

Data from three data sources.

FUNDING

This work was supported by ongoing institutional funding. No additional grants to carry out or direct this particular research were obtained.

CONFLICT OF INTEREST

The authors of this work declare that they have no conflicts of interest.

REFERENCES

1. Jørgensen, M. and Shepperd, M., A systematic review of software development cost estimation studies, *IEEE Trans. Software Eng.,* 2007, vol. 33, no. 1, pp. 33–53. https://doi.org/10.1109/TSE.2007.256943
2. Braga, P.L., Oliveira, A.L.I., and Meira, S.R.L., Software effort estimation using machine learning techniques with robust confidence intervals, *Proc. 7th Int. Conf. on Hybrid Intelligent Systems,* Kaiserslautern, 2007. https://doi.org/10.1109/his.2007.56
3. Carbonera, C.E., Farias, K., and Bischoff, V., Software development effort estimation: A systematic mapping study, *IET Res. J.,* 2020, vol. 14, no. 4, pp. 1–14. https://doi.org/10.1049/iet-sen.2018.5334
4. Fedotova, O., Teixeira, L., and Alvelos, A.H., Software effort estimation with multiple linear regression: Review and practical application, *J. Inf. Sci. Eng.,* 2013, vol. 29, no. 5, pp. 925–945.
5. Lee, T.K., Wei, K.T., and Ghani, A.A.A., Systematic literature review on effort estimation for Open Sources (OSS) web application development, *Proc. Future*

Technologies Conf. FTC 2016, San Francisco, 2016, pp. 1158–1167. https://doi.org/10.1109/FTC.2016.7821748

6. Sharma, P. and Singh, J., Systematic literature review on software effort estimation using machine learning approaches, *Proc. Int. Conf. on Next Generation Computing and Information Systems, ICNGCIS 2017,* Jammu, 2017, pp. 54–57. https://doi.org/10.1109/ICNGCIS.2017.33
7. Silhavy, R., Prokopova, Z., and Silhavy, P., Algorithmic optimization method for effort estimation, *Program. Comput. Software,* 2016, vol. 42, no. 3, pp. 161–166. https://doi.org/10.1134/S0361768816030087
8. Durán, M., Juárez-Ramírez, R., Jiménez, S., and Tona, C., User story estimation based on the complexity decomposition using Bayesian networks, *Program. Comput. Software,* 2020, vol. 46, no. 8, pp. 569–583. https://doi.org/10.1134/S0361768820080095
9. Abran, A., *Software Project Estimation: the Fundamentals for Providing High Quality Information to Decision Makers,* 1st ed., John Wiley & Sons, 2015.
10. Bilgaiyan, S., Sagnika, S., Mishra, S., and Das, M., A systematic review on software cost estimation in agile software development, *J. Eng. Sci. Technol. Rev.,* 2017, vol. 10, no. 4, pp. 51–64. https://doi.org/10.25103/jestr.104.08
11. Kinoshita, N., Monden, A., Tshunoda, M., and Yucel, Z., Predictability classification for software effort estimation, *Proc. 3rd IEEE/ACIS Int. Conf. on Big Data, Cloud Computing, Data Science and Engineering, BCD 2018,* Kanazawa, 2018, no. 1, pp. 43–48. https://doi.org/10.1109/BCD2018.2018.00015
12. Britto, R., Freitas, V., Mendes, E., and Usman, M., Effort estimation in global software development: A systematic literature review, *Proc. 9th IEEE Int. Conf. on Global Software Engineering ICGSE 2014,* Shanghai, 2014, pp. 135–144. https://doi.org/10.1109/ICGSE.2014.11
13. Valdés-Souto, F., Validation of supplier estimates using cosmic method, *Proc. Joint Conf. of the Int. Workshop on Software Measurement and the Int. Conf. on Software Process and Product Measurement, IWSM-Mensura 2019,* Haarlem, 2019, vol. 2476, pp. 15–30.
14. Yadav, N., Gupta, N., Aggarwal, M., and Yadav, A., Comparison of COSYSMO model with different software cost estimation techniques, *Proc. IEEE Int. Conf. on Issues and Challenges in Intelligent Computing Techniques, ICICT 2019,* Ghaziabad, 2019. https://doi.org/10.1109/ICICT46931.2019.8977686
15. Linda, M.C.B. and Laird, M., *Software Measurement and Estimation: A Practical Approach,* New York: Wiley, 2006.
16. Koch, S. and Mitlöhner, J., Software project effort estimation with voting rules, *Decis. Support Syst.,* 2009, vol. 46, no. 4, pp. 895–901. https://doi.org/10.1016/j.dss.2008.12.002
17. De Lucia, A., Pompella, E., and Stefanucci, S., Assessing effort estimation models for corrective maintenance through empirical studies, *Inf. Software Technol.,* 2005, vol. 47, no. 1, pp. 3–15. https://doi.org/10.1016/j.infsof.2004.05.002
18. Hill, J., Thomas, L.C., and Allen, D.E., Experts' estimates of task durations in software development projects, *Int. J. Proj. Manag.,* 2000, vol. 18, no. 1, pp. 13–21. https://doi.org/10.1016/S0263-7863(98)00062-3
19. *ISO/IEC 14143-1:2007: Information Technology – Software Measurement – Functional Size Measurement,* Technical Committee: ISO/IEC JTC 1/SC 7 Software and Systems Engineering, 2007, p. 6. https://www.iso.org/standard/38931.html.
20. *Guideline on Non-Functional & Project Requirements,* Common Software Measurement International Consortium, 2015.
21. Kitchenham, B. and Taylor, N., Software cost models, *ICL Tech. J.,* 1984, vol. 4, no. 1, pp. 73–102.
22. Valdés-Souto, F. and Naranjo-Albarrán, L., Improving the software estimation models based on functional size through validation of the assumptions behind the linear regression and the use of the confidence intervals when the reference database presents a wedge-shape form, *Program. Comput. Software,* 2021, vol. 47, no. 8, pp. 673–693. https://doi.org/10.1134/S0361768821080259
23. Shin, M. and Goel, A.L., Empirical data modeling in software engineering using radial basis functions, *IEEE Trans. Software Eng.,* 2000, vol. 26, no. 6, pp. 567–576. https://doi.org/10.1109/32.852743
24. Kitchenham, B. and Mendes, E., Why comparative effort prediction studies may be invalid, *Proc. 5th Int. Workshop on Predictive Models in Software Engineering, PROMISE 2009,* Vancouver, 2009. https://doi.org/10.1145/1540438.1540444
25. Valdés-Souto, F., Creating an estimation model from functional size approximation using the EPCU approximation approach for COSMIC (ISO 19761), in *Software Engineering: Methods, Modeling and Teaching,* Mario, C., Jaramillo, Z., Elena, C., Vanegas, D., and Charry, W.P., Eds., Bogotá, 2017, vol. 4, p. 468.
26. Abran, A., *Software Metrics and Software Metrology,* Hoboken, NJ: Wiley, 2010.
27. Morgenshtern, O., Raz, T., and Dvir, D., Factors affecting duration and effort estimation errors in software development projects, *Inf. Software Technol.,* 2007, vol. 49, no. 8, pp. 827–837. https://doi.org/10.1016/j.infsof.2006.09.006
28. Shepperd, M. and MacDonell, S., Evaluating prediction systems in software project estimation, *Inf. Software Technol.,* 2012, vol. 54, no. 8, pp. 820–827. https://doi.org/10.1016/j.infsof.2011.12.008
29. Friedman, M., The use of ranks to avoid the assumption of normality implicit in the analysis of variance, *J. Am. Stat. Assoc.,* 1937, vol. 32, no. 200, pp. 675–701. http://sci2s.ugr.es/keel/pdf/algorithm/articulo/1937-JSTOR-Friedman.pdf.
30. Lavazza, L., Accuracy evaluation of model-based COSMIC functional size estimation, *Proc. 12th Int. Conf. on Software Engineering Advances, ICSEA 2017,* Athenes, 2017, pp. 67–72.
31. Kitchenham, B.A., Pickard, L.M., MacDonell, S.G., and Shepperd, M.J., What accuracy statistics really measure, *IEE Proc. Software,* 2001, vol. 148, no. 3,

pp. 81–85.
https://doi.org/10.1049/ip-rsn:20010506

32. Foss, T., Stensrud, E., Kitchenham, B., and Myrtveit, I., A simulation study of the model evaluation criterion MMRE, *Software Eng. IEEE Trans.,* 2003, vol. 29, no. 11, pp. 985–995.
33. Myrtveit, I., Stensrud, E., and Shepperd, M., Reliability and validity in comparative studies of software prediction models, *IEEE Trans. Software Eng.,* 2005, vol. 31, no. 5, pp. 380–391.
https://doi.org/10.1109/TSE.2005.58
34. Jørgensen, M., Regression models of software development effort estimation accuracy and bias, *Empirical Software Eng. Int. J.,* 2004, vol. 9, no. 2000, pp. 297–314.
35. Sprent, N.C.S.P., *Applied Nonparametric Statistical Methods,* 4th ed., CRC Press, 2007.
36. Kruskal, W.A. and Wallis, W. H., Use of ranks in one-criterion variance analysis, *J. Am. Stat. Assoc.,* 1952, vol. 47, no. 260, pp. 583–621.
https://doi.org/10.1080/01621459.1952.10483441
37. Tukey, J.W.T.W., *Exploratory Data Analysis,* 1st ed., Addison & Wesley, 1977.
38. Wilcox, R.R., *Introduction to Robust Estimation and Hypothesis Testing,* 4th ed., Academic, 2016.
https://shop.elsevier.com/books/introduction-to-robust-estimation-and-hypothesis-testing/wilcox/978-0-12-804733-0.
39. Abran, A., et al., *Early Software Sizing with COSMIC: Experts Guide,* 2020, vol. 2020, no. International Consortium (COSMIC), pp. 1–67.
https://doi.org/10.13140/RG.2.1.4195.0567
40. Source, O.J.D., Multiple comparisons among means, *J. Am. Stat. Assoc.,* 1961, vol. 56, no. 293, pp. 52–64.
http://sci2s.ugr.es/keel/pdf/algorithm/articulo/1961-Bonferroni_Dunn-JASA.pdf.