e-ISSN 1607-3274 Радіоелектроніка, інформатика, управління. 2022. № 3
p-ISSN 2313-688X Radio Electronics, Computer Science, Control. 2022. № 3

UDC 004.412:519.237.5

NONLINEAR REGRESSION MODELS FOR ESTIMATING THE DURATION OF SOFTWARE DEVELOPMENT IN JAVA FOR PC BASED ON THE 2021 ISBSG DATA

Prykhodko S. B. – Dr. Sc., Professor, Head of the Department of Software of Automated Systems, Admiral Makarov National University of Shipbuilding, Mykolaiv, Ukraine.

Pukhalevych A. V. – PhD, Lecturer of the Department of Software of Automated Systems, Admiral Makarov National University of Shipbuilding, Mykolaiv, Ukraine.

Prykhodko K. S. – PhD, Associate Professor of the Department of Information Systems and Technologies, Admiral Makarov National University of Shipbuilding, Mykolaiv, Ukraine.

Makarova L. M. – PhD, Associate Professor, Associate Professor of the Department of Software of Automated Systems, Admiral Makarov National University of Shipbuilding, Mykolaiv, Ukraine.

ABSTRACT

Context. The problem of estimating the duration of software development in Java for personal computers (PC) is important because, first, failed duration estimating is often the main contributor to failed software projects, second, Java is a popular language, and, third, a personal computer is a widespread multi-purpose computer. The object of the study is the process of estimating the duration of software development in Java for PC. The subject of the study is the nonlinear regression models to estimate the duration of software development in Java for PC.

Objective. The goal of the work is to build nonlinear regression models for estimating the duration of software development in Java for PC based on the normalizing transformations and deleting outliers in data to increase the confidence of the estimation in comparison to the ISBSG model for the PC platform.

Method. The models, confidence, and prediction intervals of nonlinear regressions to estimate the duration of software development in Java for PC are constructed based on the normalizing transformations for non-Gaussian data with the help of appropriate techniques. The techniques to build the models, confidence, and prediction intervals of nonlinear regressions are based on normalizing transformations. Also, we apply outlier removal for model construction. In general, the above leads to a reduction of the mean magnitude of relative error, the widths of the confidence, and prediction intervals in comparison to nonlinear models constructed without outlier removal application in the model construction process.

Results. A comparison of the model based on the decimal logarithm transformation with the nonlinear regression models based on the Johnson (for the S_B family) and Box-Cox transformations as both univariate and bivariate ones has been performed.

Conclusions. The nonlinear regression model to estimate the duration of software development in Java for PC is constructed based on the decimal logarithm transformation. This model, in comparison with other nonlinear regression models, has smaller widths of the confidence and prediction intervals for effort values that are bigger than 900 person-hours. The prospects for further research may include the application of bivariate normalizing transformations and data sets to construct the nonlinear regression models for estimating the duration of software development in other languages for PC and other platforms, for example, mainframe.

KEYWORDS: duration, software development, Java, personal computer, nonlinear regression model, normalizing transformation, non-Gaussian data, ISBSG.

ABBREVIATIONS

COCOMO is a constructive cost model;

ISBSG is the International Software Benchmarking Standards Group;

KLOC is kilo lines of code (one thousand lines of code);

MMRE is a mean magnitude of relative error;

MRE is a magnitude of relative error;

PC is a personal computer;

PRED is a percentage of prediction;

SMD is a squared Mahalanobis distance.

NOMENCLATURE

$\hat{b}_0$ is an estimator of the parameter defined by the intercept of the true regression line for normalized data;

$\hat{b}_1$ is an estimator of the parameter defined by the slope of the true regression line for normalized data;

N is a number of data points;

$N\left(0,\sigma_\varepsilon^2\right)$ is a Gaussian distribution with zero mathematical expectation and variance σ_ε^2;

P is a non-Gaussian random vector;

R^2 is a multiple coefficient of determination;

T is a Gaussian random vector;

$t_{\alpha/2,N-2}$ is a quantile of student's t-distribution with $N-2$ degrees of freedom and $\alpha/2$ significance level;

X_1 is an effort of software development;

Y is the duration of software development;

Z_1 is a Gaussian variable that is obtained by transforming variable X_1;

Z_Y is a Gaussian variable that is obtained by transforming variable Y;

$\overline{Z}_Y$ is a sample mean of the Z_Y values;

$\hat{Z}_Y$ is a prediction result by linear regression equation for normalized data;

© Prykhodko S. B., Pukhalevych A. V., Prykhodko K. S., Makarova L. M., 2022
DOI 10.15588/1607-3274-2022-3-14

α is a significance level;

β_1 is a multivariate skewness;

β_2 is a multivariate kurtosis;

ε is a Gaussian random variable that defines residuals;

σ_ε is a standard deviation of ε;

$\boldsymbol{\psi}$ is a vector of bivariate normalizing transformation.

INTRODUCTION

Estimation of duration, effort, and the cost is a very important and integral part of the software development life cycle [1–3]. It is important to do an accurate estimation as much as possible because failed estimation (including duration estimation) is often the main contributor to failed software projects.

Today estimation of duration in software development is mostly based on heuristic approaches like expert judgment and planning poker. In absence of the experts for estimating, it is very difficult to estimate software development duration. That is why there is a need for algorithmic methods and mathematical models that can do accurate estimates.

For many years the most famous models are regression equations such as COCOMO and ISBSG. These models are similar in structure (both are effort dependent and constructed based on decimal logarithm). Wherein there only is one ISBSG model for estimating the duration of software development for the PC platform. However, there are no models which additionally take into account the programming language. In this paper, we demonstrated the need to take into account the programming language for the ISBSG model. We practiced the calibration of the ISBSG model using the ISBSG data set (D&E Corporate Release May 2021 R1) collected from the software development projects in Java for PC. We used the ISBSG data set because for many years the ISBSG repository is applied as a foundation of the software project estimation process [4]. Also, we constructed other models based on the normalizing transformations such as the Box-Cox and Johnson using the above data set.

The object of study is the process of estimating the duration of software development in Java for PC.

The subject of study is the regression models to estimate the duration of software development in Java for PC.

The purpose of the work is to increase the confidence in estimating the duration of software development in Java for PC.

1 PROBLEM STATEMENT

Suppose given the original sample as the bivariate non-Gaussian data set: actual duration (in months) Y and effort (in person-hours) X_1 of software development in Java for PC. Suppose that there is a mutually inverse normalizing transformation of non-Gaussian random vector $\mathbf{P} = \{Y, X_1\}^T$ to Gaussian random vector $\mathbf{T} = \{Z_Y, Z_1\}^T$ is given by

$$\mathbf{T} = \boldsymbol{\psi}(\mathbf{P}) \tag{1}$$

and the inverse transformation for (1)

$$\mathbf{P} = \boldsymbol{\psi}^{-1}(\mathbf{T}). \tag{2}$$

It is required to build the nonlinear regression model in the form $Y = Y(X_1, \varepsilon)$ based on transformations (1) and (2).

2 REVIEW OF THE LITERATURE

Although the first models for estimating the duration of software development were built in the 1970–1980 years [1, 5], research in this area is still ongoing [4, 6–10].

Most often these models enable estimating the duration of software development depending on the development effort. Building such models requires the presence of corresponding datasets. Firstly it was government organization datasets (NASA etc.). For at least 25 years many such researchers are used data from the different ISBSG repository releases [6–10].

The COCOMO models were built using project size as the data clustering criteria [1]. Software development projects were split by their size into 3 types: organic (2–50 KLOC), semi-detached (50–300 KLOC), and embedded (larger than 300 KLOC). Then each of these types was built in separate models.

The ISBSG models are similar in structure to COCOMO models the only difference is that they were built for such platforms as mainframe, mid-range, and personal computers based on the 1996 ISBSG repository data.

In all models from [1, 2, 6] the decimal logarithm transformation was used to normalize empirical data. But as it was clear from [6], the above transformation is not always acceptable for empirical data normalization. In [6] a linear regression was performed on the Log10-transformed values of duration and effort for the 39 PC software development projects $R^2 = 0.140$. It is very low and means that there is no correlation between dependent and independent variables.

In the nonlinear regression model for estimating the duration of software development for PC [7], the Johnson univariate transformation was used to normalize empirical data values of duration and effort. This transformation enables to build of valid models in some cases but as will be shown in this research this transformation gives average model quality with the 2021 ISBSG repository data for software developed in Java for PC. Therefore, it is also required to apply bivariate transformation and remove outliers from the empirical data to build a high-quality model according to [11].

© Prykhodko S. B., Pukhalevych A. V., Prykhodko K. S., Makarova L. M., 2022
DOI 10.15588/1607-3274-2022-3-14

A normalizing transformation is often a good way to construct nonlinear regression models [11–17]. According to [14], transformations are made for essentially four purposes, two of which are: firstly, to obtain approximate normality for the distribution of the error term (residuals), secondly, to transform the response and/or the predictor in such a way that the strength of the linear relationship between new variables (normalized variables) is better than the linear relationship between initial dependent and independent variables.

According to [11], there may be data sets on which the results of building nonlinear regression models depend, firstly, which normalizing transformation is used, univariate, or multivariate, and, secondly, are there any outliers in the data set. That is why in [11] the technique was considered to build nonlinear regression models based on the multivariate normalizing transformations and prediction intervals. In this technique, in addition to the technique for detecting outliers in multivariate non-Gaussian data [18], the prediction intervals of nonlinear regressions are used to detect the outliers in the process of constructing the nonlinear regression models. We apply the above technique [11] for building the nonlinear regression models with one predictor (effort) to estimate the duration of software development in Java for PC.

3 MATERIALS AND METHODS

According to [11], the technique to build nonlinear regression models based on the normalizing transformations and prediction intervals consists of four steps. In the first step, multivariate non-Gaussian data are normalized using a multivariate normalizing transformation (1).

In the second step, the nonlinear regression model is constructed based on the multivariate normalizing transformation (1). Before that, we first determine whether one data point of a multivariate non-Gaussian data set is a multidimensional outlier. To do this, we apply the statistical technique based on the normalizing transformations and the Mahalanobis squared distance (MSD) as in [18, 19]. If there is a two-dimensional outlier in a bivariate non-Gaussian data set, then we discard the one, and return to step 1, else build the linear regression model for normalized data based on the transformation (1) in the form

$$Z_Y = \hat{Z}_Y + \varepsilon = \hat{b}_0 + \hat{b}_1 Z_1 + \varepsilon , \qquad (3)$$

ε is a Gaussian random variable that defines residuals, $\varepsilon \sim N\left(0, \sigma_\varepsilon^2\right)$.

After that, the nonlinear regression model is built based on the linear regression model (3) and the transformations (1) and (2) as

$$Y = \psi_Y^{-1}\left(\hat{Z}_Y + \varepsilon\right) . \qquad (4)$$

In the third step, the prediction interval of nonlinear regression is defined [11]

$$\psi_Y^{-1}\left(\hat{Z}_{Y_i} \pm t_{\alpha/2, N-2} S_Z \sqrt{1 + \frac{1}{N} + \frac{\left(Z_{1_i} - \overline{Z}_1\right)^2}{S_{Z_1 Z_1}}} \right) , \qquad (5)$$

where $t_{\alpha/2,N-2}$ is a student's t-distribution quantile with $\alpha/2$ significance level and $N-2$ degrees of freedom; $S_Z^2 = \frac{1}{N-2}\sum_{i=1}^{N}\left(Z_{Y_i} - \hat{Z}_{Y_i}\right)^2$, $S_{Z_1 Z_1} = \sum_{i=1}^{N}\left(Z_{1_i} - \overline{Z}_1\right)^2$; $\overline{Z}_1 = \frac{1}{N}\sum_{i=1}^{N} Z_{1_i}$.

In the fourth step, we check if there are data that are out of the bounds of the prediction interval. And if we detect the outliers, we discard them and repeat all the steps starting with the first for new data without discarded outliers, else nonlinear regression model construction is completed.

To normalize the data according to (1), we applied the decimal logarithm transformation with components Z_1

$$Z_1 = \lg X_1 \qquad (6)$$

and Z_Y

$$Z_Y = \lg Y . \qquad (7)$$

Also, to normalize the data, we used the univariate and bivariate Box-Cox transformations [16] with components Z_1

$$Z_1 = x(\lambda_1) = \begin{cases} \left(X_1^{\lambda_1} - 1\right) \big/ \lambda_1, & \text{if} \quad \lambda_1 \neq 0; \\ \ln\left(X_1\right), & \text{if} \quad \lambda_1 = 0 \end{cases} \qquad (8)$$

and Z_Y, which is defined analogously to (8) with the only difference that instead of Z_1, X_1, and λ_1 should be put respectively Z_Y, Y, and λ_Y. Here Z_1 and Z_Y are Gaussian variables, λ_1 and λ_Y parameters of the bivariate Box-Cox transformation.

Furthermore, to normalize the data, we used the univariate and bivariate Jonson transformations for the S_B family [11] with component Z_1

$$Z_1 = \gamma_1 + \eta_1 \ln \frac{X_1 - \varphi_1}{\varphi_1 + \lambda_1 - X_1} \qquad (9)$$

and Z_Y, which is defined analogously to (9) with the only difference that instead of Z_1, X_1, γ_1, η_1, φ_1, and λ_1 should be put respectively Z_Y, Y, γ_Y, η_Y, φ_Y, and λ_Y. Here Z_1 and Z_Y are Gaussian variables with zero mathematical expectation and unit variance; γ_Y, η_Y,

© Prykhodko S. B., Pukhalevych A. V., Prykhodko K. S., Makarova L. M., 2022
DOI 10.15588/1607-3274-2022-3-14

φ_Y, λ_Y, γ_1, η_1, φ_1, and λ_1 are parameters of the Johnson transformation for the S_B family.

The nonlinear regression model based on the linear regression model (3) for the normalized data and the decimal logarithm transformation for (6) and (7) has the form

$$Y = 10^{\varepsilon+\hat{b}_0} X_1^{\hat{b}_1}. \quad (10)$$

The nonlinear regression model based on the bivariate Box-Cox transformation has the form [20]

$$Y = \left[\hat{\lambda}_Y\left(\hat{Z}_Y + \varepsilon\right)+1\right]^{1/\hat{\lambda}_Y}. \quad (11)$$

According to [20], the nonlinear regression model based on the Johnson bivariate transformation for the S_B family has the form

$$Y = \hat{\phi}_Y + \hat{\lambda}_Y \Big/ \left\{1+\exp\left[-\left(\hat{Z}_Y + \varepsilon - \hat{\gamma}_Y\right)\Big/\hat{\eta}_Y\right]\right\}. \quad (12)$$

In (10)–(12) as and in (3), ε is a Gaussian random variable which defines residuals, $\varepsilon \sim N\left(0, \sigma_\varepsilon^2\right)$.

The confidence interval of nonlinear regression is defined analogously to (5) with the only difference that in the sum under the square root, there will not be leading 1.

4 EXPERIMENTS

Before building a nonlinear regression model based on the multivariate normalizing transformation, we constructed a nonlinear regression equation to estimate the duration Y (in months) of software development for the PC platform depending on the effort X_1 (in person-hours) based on the decimal logarithm transformation of 243 software projects data with Data Quality Rating A from the 2021 ISBSG database (see Fig. 1).

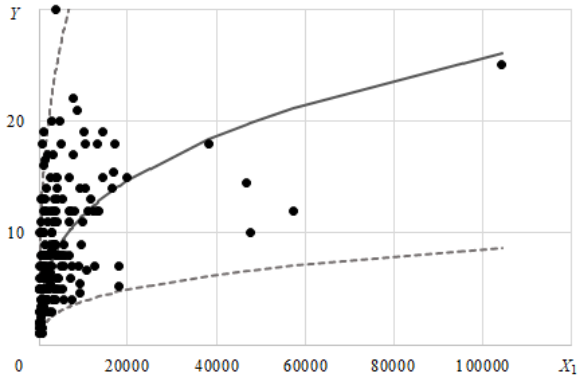

Figure 1 – Nonlinear regression (solid line) and its prediction intervals (dash lines) of the duration depending on the effort, which was constructed by the decimal logarithm using 243 software projects data (dots) with Quality Rating A (highest) from the ISBSG database (D&E Corporate Release May 2021 R1)

Fig. 1 contains nonlinear regression (solid line) for which the equation is:

$$Y = 0.4902 X_1^{0.3437}. \quad (13)$$

Also, Fig. 1 contains prediction intervals bounds (dash lines) of nonlinear regression of the duration depending on the effort, which was constructed using the decimal logarithm (Log10) by (5) for a significance level of 0.05.

The values of R^2, MMRE, and PRED(0.25) equal respectively 0.2971, 0.4840, and 0.3457 for equation (13). These values are less than acceptable ones and indicate the unsatisfactory accuracy of duration prediction by equation (1). That is why we apply the appropriate technique [11] to build a nonlinear regression model for estimating the duration of software development in Java for PC.

To construct a nonlinear regression model for estimating the duration of software development in Java for PC we use the above technique for the 39 software projects data with Data Quality Rating A from the ISBSG database (D&E Corporate Release May 2021 R1). The above data are shown in Fig. 2 as dots.

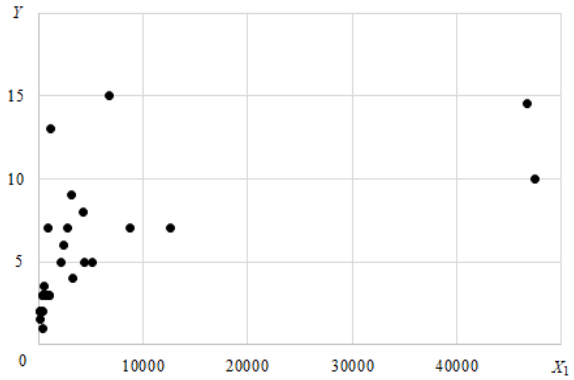

Figure 2 – Scatter plot of effort X_1 vs. duration Y for 39 software projects in Java for PC

We checked the bivariate data from Fig. 2 for multivariate outliers. But before that, we tested the normality of multivariate data from Fig. 2 because well-known statistical methods (for example, multivariate outlier detection based on the squared Mahalanobis distance (SMD)) are used to detect outliers in multivariate data under the assumption that the data is described by a multivariate Gaussian distribution [16, 18, 19]. We applied a multivariate normality test proposed by Mardia and based on measures of the multivariate skewness β_1 and kurtosis β_2 [21, 22]. According to this test, the distribution of bivariate data from Fig. 2 is not Gaussian since the test statistic for multivariate skewness $N\beta_1/6$ of this data, which equals 119.61, is greater than the quantile of the Chi-Square distribution, which is 14.86 for 4 degrees of freedom and 0.005 significance level. Similarly, the test statistic for multivariate kurtosis β_2, which equals 24.60,

© Prykhodko S. B., Pukhalevych A. V., Prykhodko K. S., Makarova L. M., 2022
DOI 10.15588/1607-3274-2022-3-14

is greater than the value of the Gaussian distribution quantile, which is 11.30 for 8 mean, 1.641 variance, and 0.005 significance level.

Therefore, we used the statistical technique [18] to detect multivariate outliers in the bivariate non-Gaussian data from Fig. 2 based on the multivariate normalizing transformations and the SMD for normalized data. To normalize the data from Fig. 2, we applied three univariate and two bivariate transformations (see Table 1).

The parameter estimates of the univariate and bivariate Box-Cox transformations for the data from Fig. 2 are calculated by the maximum likelihood method according to [16]. The parameter estimates of the univariate Box-Cox transformation are $\hat{\lambda}_Y = -0.295079$ and $\hat{\lambda}_1 = -0.244234$. The parameter estimates of the bivariate Box-Cox transformation are $\hat{\lambda}_Y = -0.207256$ and $\hat{\lambda}_1 = -0.212036$.

Table 1 – SMD values for normalized data

No	Project ID	Univariate			Bivariate	
		Log10	Box-Cox	Johnson	Box-Cox	Johnson
1	10248	1.37	1.55	1.04	1.60	1.27
2	10868	8.74	**14.69**	**14.02**	**12.95**	10.56
3	11641	**11.81**	8.18	8.20	9.10	10.23
4	11802	1.12	1.26	1.12	1.23	1.19
5	12636	5.60	3.98	4.78	4.42	5.30
6	12857	0.98	0.98	0.89	0.97	0.96
7	14345	0.43	0.39	0.26	0.40	0.31
8	14487	2.33	1.90	1.93	1.98	2.20
9	14883	0.50	0.49	0.31	0.49	0.38
10	14937	6.61	4.09	5.57	4.39	5.73
11	14953	0.52	0.61	0.51	0.59	0.53
12	16032	1.24	1.61	1.23	1.57	1.30
13	18271	0.15	0.13	0.12	0.15	0.16
14	19256	0.73	0.85	0.49	0.83	0.59
15	21372	0.42	0.35	0.21	0.34	0.27
16	21719	0.29	0.21	0.16	0.22	0.20
17	22359	1.86	1.92	1.36	2.01	1.68
18	22404	3.43	2.63	2.52	2.77	2.91
19	23094	1.11	1.28	1.05	1.26	1.11
20	23265	0.28	0.20	0.16	0.21	0.20
21	24483	2.23	2.01	1.76	2.07	1.98
22	25081	0.16	0.09	0.10	0.11	0.13
23	25342	1.54	1.86	1.13	1.93	1.47
24	25480	0.16	0.09	0.10	0.11	0.13
25	25663	1.49	2.26	1.66	2.17	1.76
26	25931	0.41	0.63	0.37	0.65	0.50
27	26422	0.25	0.17	0.14	0.18	0.17
28	26695	2.17	2.77	2.46	2.62	2.34
29	28504	0.28	0.42	0.26	0.43	0.34
30	28519	2.73	5.90	9.75	5.45	6.30
31	29310	7.92	4.10	6.06	4.45	6.44
32	29311	1.37	1.37	1.27	1.38	1.35
33	29398	1.83	3.18	2.49	3.00	2.52
34	29471	0.14	0.12	0.11	0.13	0.14
35	29537	0.23	0.43	0.27	0.41	0.27
36	30243	0.40	0.61	0.36	0.63	0.48
37	30658	0.18	0.24	0.17	0.25	0.22
38	31895	3.86	3.17	2.57	3.31	3.26
39	31999	1.11	1.28	1.05	1.26	1.11

The parameter estimates of the univariate and bivariate Jonson transformation for the S_B family for the data from Fig. 2 are calculated by the maximum likelihood method according to [20]. The parameter estimates of the univariate Jonson transformation for the S_B family are $\hat{\gamma}_Y = 3.8025$, $\hat{\gamma}_1 = 3.02479$, $\hat{\eta}_Y = 1.4727$, $\hat{\eta}_1 = 0.59352$, $\hat{\varphi}_Y = 0.65925$, $\hat{\varphi}_1 = 97.897$, $\hat{\lambda}_Y = 76.853$, and $\hat{\lambda}_1 = 207294.1$. The parameter estimates of the bivariate Jonson transformation for the S_B family are $\hat{\gamma}_Y = 10.939$, $\hat{\gamma}_1 = 4.42142$, $\hat{\eta}_Y = 1.19033$, $\hat{\eta}_1 = 0.54626$, $\hat{\varphi}_Y = 0.23642$, $\hat{\varphi}_1 = 81.576$, $\hat{\lambda}_Y = 5937.939$, and $\hat{\lambda}_1 = 1484909.7$.

Table 1 contains the SMD for normalized data. The SMD values from Table 1 indicate there is one multivariate outlier in bivariate non-Gaussian data for four transformations (all univariate transformations and the bivariate Box-Cox transformation) since the SMD values for row 3 for decimal logarithm and row 2 for two univariate transformations (Box-Cox and Jonson) and the bivariate Box-Cox transformation are greater than the quantile of the Chi-Square distribution, which equals to 10.60 for the 0.005 significance level and 2 degrees of freedom. In Table 1, the SMD values, which are greater than the above quantile, are highlighted in bold.

For example, a scatter plot of normalized effort Z_1 vs. normalized duration Z_Y (using the bivariate Box-Cox transformation) for the data from Fig. 2 is shown in Fig. 3. Here the above outlier (Project 10868) is marked as an "outlier".

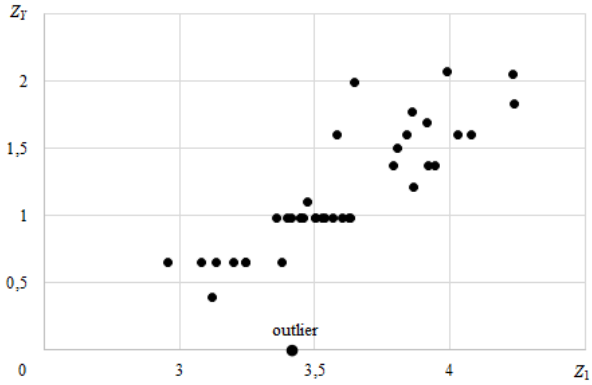

Figure 3 – Scatter plot of normalized effort Z_1 vs. normalized duration Z_Y (using the bivariate Box-Cox transformation) for the data from Fig. 2

Only the SMD values from Table 1 for bivariate Jonson transformations for the S_B family indicate there are no multivariate outliers in bivariate non-Gaussian data from Fig. 2 since all SMD values, in this case, are less than the above quantile value.

The reason for such different results in outlier detection is that only the data normalized using the bivariate Jonson transformation for the S_B family passes a multivariate normality test proposed by Mardia [21]. As a note, the above, Mardia's test is based on measures of the multivariate skewness β_1 and kurtosis β_2 [21].

© Prykhodko S. B., Pukhalevych A. V., Prykhodko K. S., Makarova L. M., 2022
DOI 10.15588/1607-3274-2022-3-14

According to Mardia's test, the bivariate distribution of data (from Fig. 2) normalized using the bivariate Jonson transformation for the S_B family is approximately Gaussian since the test statistic for multivariate skewness $N\beta_1/6$ of this data, which equals 2.08, is less than the quantile of the Chi-Square distribution, which is 14.86 for 4 degrees of freedom and 0.005 significance level. Also, the test statistic for multivariate kurtosis β_2, which equals 10.71, is less than the value of the Gaussian distribution quantile, which is 11.30 for 8 mean, 1.641 variance, and 0.005 significance level.

Therefore, we decide, that there are no multivariate outliers in bivariate non-Gaussian data from Fig. 2 (39 data points). And we go to step 2 of the first iteration.

We constructed the nonlinear regression model (10), for which the estimate $\hat{\sigma}_\varepsilon$ is 0.1552, parameters estimates are $\hat{b}_0 = -0.500291$ and $\hat{b}_1 = 0.357533$.

Next, we calculated the nonlinear regression prediction interval by (5) for a significance level of 0.05. In the first iteration, $t_{\alpha/2,N-2} = 2.026$; $S_Z = 0.15726$; $\overline{Z}_1 = 3.025$; $S_{Z_1Z_1} = 15.968$ for the data normalized by the Log10 transformation of 39 data points from Fig. 2.

There are two outliers (data for software projects 10868 and 11641) since their Y values are out of the prediction interval computed by (5) for a significance level of 0.05. We discarded data of software projects 10868 and 11641. The first iteration is completed. The above 37 data points are shown in Fig. 4.

In the second iteration, there are no multivariate outliers in bivariate non-Gaussian data from Fig. 4 (37 data points). And we go to step 2 of the second iteration.

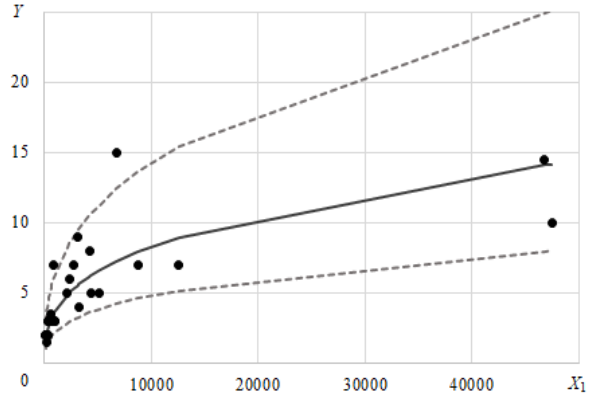

Figure 4 – Nonlinear regression $\hat{Y}$ (solid line) and its prediction intervals (dash lines) of the duration depending on the effort, which is constructed by the decimal logarithm transformation of 37 data points

We constructed the nonlinear regression model (10), for which the estimate $\hat{\sigma}_\varepsilon$ is 0.1104, parameters estimates are $\hat{b}_0 = -0.468116$ and $\hat{b}_1 = 0.346194$.

Next, we calculated the nonlinear regression prediction interval by (5) for a significance level of 0.05. In the second iteration, $t_{\alpha/2,N-2} = 2.030$; $S_Z = 0.1120$; $\overline{Z}_1 = 3.035$; $S_{Z_1Z_1} = 15.816$ for the data normalized by the Log10 transformation of 37 data points from Fig. 4.

There are two outliers (data for software projects 12636 and 31895) since their Y values are out of the prediction interval computed by (5) for a significance level of 0.05. We discarded data from software projects 12636 and 31895. The second iteration is completed.

In the third iteration, we used data from the remaining 35 projects (see Fig. 5). There are no multivariate outliers in bivariate non-Gaussian data from Fig. 5 (35 data points). And we go to step 2 of the third iteration.

Next, we used 35 data points from Fig. 5 to construct the model in form (10) with the following parameters estimates: $\hat{b}_0 = -0.439718$, $\hat{b}_1 = 0.330913$, $\hat{\sigma}_\varepsilon = 0.0828$.

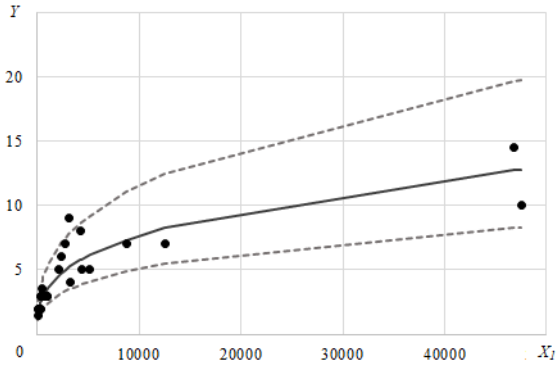

Figure 5 – Nonlinear regression $\hat{Y}$ (solid line) and its prediction intervals (dash lines) of the duration depending on the effort, which is constructed by the decimal logarithm transformation of 35 data points

After constructing a model (10), we have to find the nonlinear regression prediction interval by (5) for a significance level of 0.05 (see Fig. 5). In the third iteration, $t_{\alpha/2,N-2} = 2.035$; $S_Z = 0.0840$; $\overline{Z}_1 = 3.016$; $S_{Z_1Z_1} = 15.158$ for the data normalized by the Log10 transformation of 35 data points from Fig. 5.

There is one outlier (data for software project 14487) since its Y value is out of the prediction interval computed by (5) for a significance level of 0.05. We discarded data from software project 14487. The third iteration is completed.

In the fourth iteration, we used data from the remaining 34 projects (see Fig. 6). There are no multivariate outliers in bivariate non-Gaussian data from Fig. 6 (34 data points). And we go to step 2 of the fourth iteration.

We used 34 data points from Fig. 6 to construct the model in form (10) with the following parameters estimates: $\hat{b}_0 = -0.423179$, $\hat{b}_1 = 0.323072$, $\hat{\sigma}_\varepsilon = 0.07253$.

Next, we calculated the nonlinear regression prediction interval by (5) for a significance level of 0.05. In the fourth iteration, $t_{\alpha/2,N-2} = 2.037$; $S_Z = 0.07366$;

© Prykhodko S. B., Pukhalevych A. V., Prykhodko K. S., Makarova L. M., 2022
DOI 10.15588/1607-3274-2022-3-14

$\overline{Z}_1 = 3.002$; $S_{Z_1Z_1} = 14.923$ for the data normalized by the Log10 transformation of 34 data points from Fig. 6.

There are no outliers since all Y values are not out of the bounds of the prediction interval computed by (5) for a significance level of 0.05. The model construction is completed.

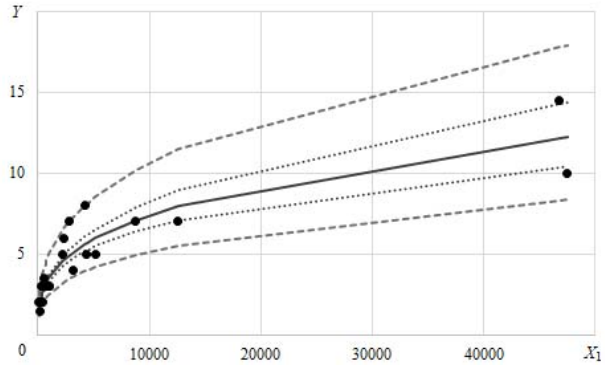

Figure 6 – Nonlinear regression $\hat{Y}$ (solid line), its confidence (dot lines) and prediction (dash lines) intervals of the duration depending on the effort, which is constructed by the decimal logarithm transformation of 34 data points

Also, we calculated the confidence intervals of nonlinear regression $\hat{Y}$ constructed by the decimal logarithm transformation of 34 data points (see Fig. 6).

The computer program implementing the constructed models (10), (11), and (12) was developed to conduct experiments. The program was written in the sci-language for the Scilab system. Scilab (https://www.scilab.org/) is free and open-source software, the alternative to commercial packages for system modeling and simulation packages such as MATLAB and MATRIXx [23].

5 RESULTS

The prediction results $\hat{Y}$ (solid line) of nonlinear regression models (11) and (12), its confidence (dot lines) and prediction (dash lines) intervals of the duration (in months) depending on the effort (in person-hours) are defined for both univariate and multivariate transformations (see figures 7–10) to compare with prediction results for model (10).

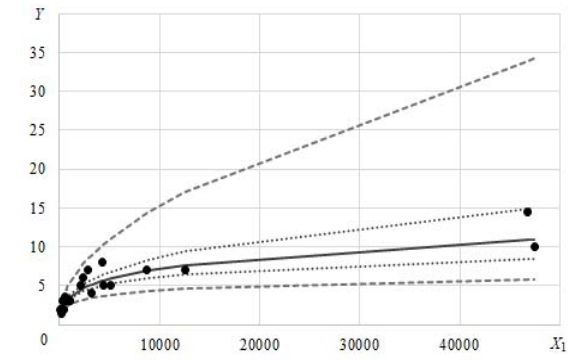

Figure 7 – Nonlinear regression $\hat{Y}$ (solid line), its confidence (dot lines) and prediction (dash lines) intervals of the duration depending on the effort, which is constructed by univariate Box-Cox' transformation of 34 data points

To evaluate the prediction accuracy of the nonlinear regression models we applied the metrics R^2, MMRE, and PRED(0.25). MMRE and PRED(0.25) are accepted as standard evaluations of prediction results by regression models.

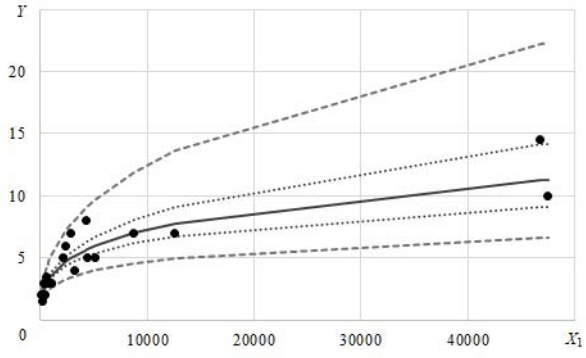

Figure 8 – Nonlinear regression $\hat{Y}$ (solid line), its confidence (dot lines) and prediction (dash lines) intervals of the duration depending on the effort, which is constructed by bivariate Box-Cox' transformation of 34 data points

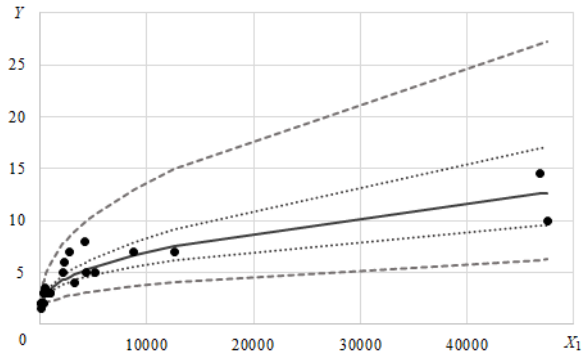

Figure 9 – Nonlinear regression $\hat{Y}$ (solid line), its confidence (dot lines) and prediction (dash lines) intervals of the duration depending on the effort, which is constructed by univariate Johnson' transformation of 34 data points

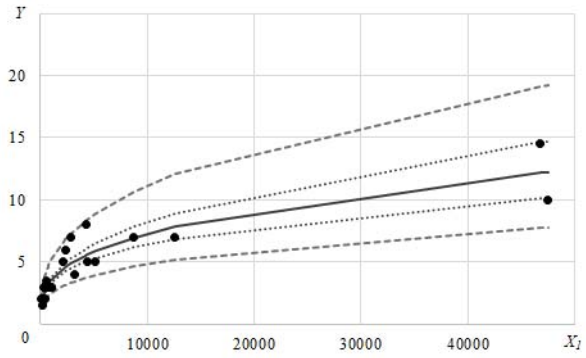

Figure 10 – Nonlinear regression $\hat{Y}$ (solid line), its confidence (dot lines) and prediction (dash lines) intervals of the duration depending on the effort, which constructed by bivariate Johnson' transformation of 34 data points

© Prykhodko S. B., Pukhalevych A. V., Prykhodko K. S., Makarova L. M., 2022
DOI 10.15588/1607-3274-2022-3-14

These metrics are applied in software engineering too [24, 25]. The acceptable values of MMRE and PRED(0.25) are not more than 0.25 and not less than 0.75 respectively. The values of R^2, MMRE and PRED(0.25) are shown in Table 2 for models (10)–(12) for both univariate and multivariate transformations. The values of these metrics are acceptable and approximately the same for all models. These values indicate good prediction accuracy of the models (10)–(12) for estimating the duration of software development in Java for PC.

Table 2 – The prediction accuracy metrics of the nonlinear regression models

Metrics	Univariate			Bivariate	
	Log10	Box-Cox	Johnson	Box-Cox	Johnson
R^2	0.8817	0.8704	0.8763	0.8709	0.8807
MMR_{min}	0.0010	0.0008	0.0006	0.0086	0.0001
MMR_{max}	0.3147	0.3001	0.2953	0.3539	0.3108
MMRE	0.1352	0.1333	0.1346	0.1356	0.1353
PRED(0.25)	0.8529	0.8529	0.8529	0.8235	0.8529

Also, Table 2 contains minimum and maximum values of MRE denoted MMR_{min} and MMR_{max}, respectively.

The model (12) based on the Johnson bivariate transformation for the S_B family has smaller MRE values for bigger numbers of data points in comparison to other models. Such, the MRE values for the model (12) based on the Johnson bivariate transformation are smaller than for the model (11) with parameters estimates for both the univariate and bivariate Box-Cox transformations for 21 from 34 data points. The MRE values for the model (12) based on the Johnson bivariate transformation are smaller than for the model (10) for 18 from 34 data points. The MRE values for the model (12) based on the Johnson bivariate transformation are smaller than for the model (12) with parameter estimates for the Johnson univariate transformation for 19 from 34 data points. Also, the last result indicates the advantage of using the bivariate transformation in comparison to the univariate one.

6 DISCUSSION

We apply bivariate normalizing transformations to build the nonlinear regression model for estimating the duration of software development in Java for PC by appropriate techniques [11] since the error distribution of the linear regression model is not Gaussian what the chi-squared test result indicates. Also, there are no outliers in the data. Moreover, the bivariate distribution of the data is not Gaussian which the Mardia multivariate normality test based on measures of the multivariate skewness and kurtosis indicates. Because we use the statistical technique [18] to detect multivariate outliers in the bivariate non-Gaussian data based on the bivariate normalizing transformations and the SMD for normalized data. Note, that we have other bivariate outliers for the data from Table 1 without applying normalization compared to outlier detection results using the above technique [18].

Also note that in our case, the poor normalization of bivariate non-Gaussian data using the Box-Cox and Johnson univariate transformations lead to an increase in the widths of the confidence and prediction intervals of nonlinear regression for a larger number of data rows compared to the Box-Cox and Johnson bivariate transformations. The above indicates the advantage of using the bivariate transformation in comparison to the univariate one.

The nonlinear regression model (10), in comparison with other nonlinear regression models (11) and (12), has smaller widths of the confidence and prediction intervals for effort values that are bigger than 900 person-hours. These results and the values of the prediction accuracy metrics from Table 2 indicate the preference for using a more simple model (10) for estimating the duration of software development in Java for PC.

CONCLUSIONS

The important problem of increase of confidence in estimating the duration of software development in Java for PC is solved.

The scientific novelty of obtained results is that nonlinear regression models to estimate the duration of software development in Java for PC are firstly constructed based on the Box-Cox and Johnson bivariate transformations. These models, in comparison with other nonlinear regression models, have smaller widths of the confidence and prediction intervals for effort values that are smaller than 900 person-hours.

The practical significance of obtained results is that the software realizing the constructed model is developed in the sci-language for Scilab. The experimental results allow for the recommendation of the constructed model for use in practice.

Prospects for further research may include the application of bivariate normalizing transformations and data sets to construct the nonlinear regression models for estimating the duration of software development in other languages for PC and other platforms, for example, mainframe.

ACKNOWLEDGEMENTS

This research was made possible by the ISBSG data set (D&E Corporate Release May 2021 R1). ISBSG (www.isbsg.org) provided the repository data subscription for Admiral Makarov National University of Shipbuilding at a heavily discounted price for academic purposes.

REFERENCES

1. Boehm B. W. Software engineering economics. Englewood Cliffs, NJ, Prentice Hall, 1981, 768 p.
2. Boehm B. W., Abts C., Brown A. W. et al. Software cost estimation with COCOMO II. Upper Saddle River, NJ: Prentice Hall PTR, 2000, 506 p.
3. Owais M., Ramakishore R. Effort, duration and cost estimation in agile software development, *2016 Ninth International Conference on Contemporary Computing (IC3),* 2016, pp. 1–5, DOI: 10.1109/IC3.2016.7880216.
4. Abran A. Data collection and industry standards: the ISBSG repository, *Software Project Estimation: The Fundamentals for Providing High Quality Information to Decision Makers,*

© Prykhodko S. B., Pukhalevych A. V., Prykhodko K. S., Makarova L. M., 2022
DOI 10.15588/1607-3274-2022-3-14

IEEE, 2015, pp. 161–184, DOI: 10.1002/9781118959312.ch8.
5. Putnam L. H. A general empirical solution to the macro-software sizing and estimating problem, *IEEE Transactions on Software Engineering,* 1978, Vol. 4, No. 2, July, pp. 345–361.
6. Oligny S., Bourque P. , Abran A. , Fournier B. Exploring the relation between effort and duration in software engineering projects, *Proceedings of the World Computer Congress,* Aug. 2000, P. 175–178.
7. Prykhodko S. B., Pukhalevich A. V. Developing PC Software Project Duration Model based on Johnson transformation, *Proceedings of the 12th International Conference Modern Problems of Radio Engineering, Telecommunications and Computer Science TCSET'2014, Lviv-Slavske, Ukraine*. Lviv, Polytechnic National University, 2014, pp. 114–116.
8. Prykhodko S.B., Pukhalevich A. V. Confidence interval estimation of PC software project duration regression based on Johnson transformation, *Radioelectronic and Computer Systems*. Kharkiv, 2014, No. 2 (66), pp. 104–107. ISSN: 1814-4225
9. López-Martín C., Abran A. Neural networks for predicting the duration of new software projects, *Journal of Systems and Software,* 2015, Vol. 101, pp. 127–135. DOI: 10.1016/J.JSS.2014.12.002
10. Pospieszny P., Czarnacka-Chrobot B., Kobylinski A. An effective approach for software project effort and duration estimation with machine learning algorithms, *Journal of Systems and Software,* 2018, pp. 184–196. DOI: 10.1016/J.JSS.2017.11.066
11. Prykhodko S., Prykhodko N. Mathematical modeling of non-Gaussian dependent random variables by nonlinear regression models based on the multivariate normalizing transformations, *Mathematical Modeling and Simulation of Systems : 15th International Scientific-practical Conference MODS'2020*. Chernihiv, Ukraine, June 29 – July 01, 2020, selected papers. Springer, Cham, 2021, P. 166–174. (Advances in Intelligent Systems and Computing, Vol. 1265). DOI: 10.1007/978-3-030-58124-4_16
12. Bates D. M., Watts D. G. Nonlinear regression analysis and its applications. New York, John Wiley & Sons, 1988, 384 p. DOI:10.1002/9780470316757
13. Seber G.A.F., C. J. Wild Nonlinear regression. New York, John Wiley & Sons, 1989, 768 p. DOI: 10.1002/0471725315
14. Ryan T. P. Modern regression methods. New York, John Wiley & Sons, 1997, 529 p. DOI: 10.1002/9780470382806
15. Drapper N. R., Smith H. Applied regression analysis. New York, John Wiley & Sons, 1998, 736 p.
16. Johnson R. A., Wichern D. W. Applied multivariate statistical analysis, Pearson Prentice Hall, 2007, 800 p.
17. Chatterjee S., Simonoff J. S. Handbook of regression analysis. New York, John Wiley & Sons, 2013, 236 p. DOI: 10.1002/9781118532843
18. Prykhodko S., Prykhodko N., Makarova L., et al. Detecting Outliers in Multivariate Non-Gaussian Data on the basis of Normalizing Transformations, *Electrical and Computer Engineering : the 2017 IEEE First Ukraine Conference (UKRCON) «Celebrating 25 Years of IEEE Ukraine Section», Kyiv, Ukraine, May 29 – June 2, 2017 : proceedings.* Kyiv, IEEE, 2017, pp. 846–849. DOI: 10.1109/UKRCON.2017.8100366
19. Prykhodko S., Prykhodko N., Makarova L. et al. Application of the Squared Mahalanobis Distance for Detecting Outliers in Multivariate Non-Gaussian Data, *Radioelectronics, Telecommunications and Computer Engineering : 14th International Conference on Advanced Trends (TCSET).* Lviv-Slavske, Ukraine, February 20–24, 2018 : proceedings, pp. 962–965. DOI: 10.1109/TCSET.2018.8336353
20. Prykhodko S., Prykhodko N., Knyrik K. Estimating the efforts of mobile application development in the planning phase using nonlinear regression analysis, *Applied Computer Systems*, 2020, Vol. 25, No. 2, pp. 172–179. DOI: 10.2478/acss-2020-0019
21. Mardia K.V. Measures of multivariate skewness and kurtosis with applications, *Biometrika,* 1970, Vol. 57, pp. 519–530. DOI: 10.1093/biomet/57.3.519
22. Mardia K.V. Applications of some measures of multivariate skewness and kurtosis in testing normality and robustness studies, *Sankhya: The Indian Journal of Statistics, Series B (1960–2002),* 1974, Vol. 36, Issue 2, pp. 115–128.
23. Campbell S. L., Chancelier J.-P., Nikoukhah R.Modeling and simulation in Scilab/Scicos. Springer, 2005, 313 p.
24. Foss T., Stensrud E., Kitchenham B., Myrtveit I. A simulation study of the model evaluation criterion MMRE, *IEEE Transactions on software engineering,* 2003, Vol. 29, Issue 11, pp. 985–995. DOI: 10.1109/TSE.2003.1245300
25. Port D., Korte M. Comparative studies of the model evaluation criterions MMRE and PRED in software cost estimation research, *Empirical Software Engineering and Measurement, the 2nd ACM-IEEE International Symposium ESEM, Kaiserslautern, Germany, October, 2008 : proceedings.* New York, ACM, 2008, pp. 51–60

Received 01.07.2022.
Accepted 27.08.2022.

УДК 004.412:519.237.5

НЕЛІНІЙНІ РЕГРЕСІЙНІ МОДЕЛІ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ НА JAVA ДЛЯ ПК ЗА ДАНИМИ ISBSG 2021 РОКУ

Приходько С. Б. – д-р техн. наук, професор, завідувач кафедри програмного забезпечення автоматизованих систем Національного університету кораблебудування імені адмірала Макарова, Миколаїв, Україна.

Пухалевич А. В. – канд. техн. наук, викладач кафедри програмного забезпечення автоматизованих систем Національного університету кораблебудування імені адмірала Макарова, Миколаїв, Україна.

Приходько К. С. – канд. техн. наук, доцент кафедри інформаційних систем і технологій Національного університету кораблебудування імені адмірала Макарова, Миколаїв, Україна.

Макарова Л. М. – канд. техн. наук, доцент кафедри програмного забезпечення автоматизованих систем Національного університету кораблебудування імені адмірала Макарова, Миколаїв, Україна.

© Prykhodko S. B., Pukhalevych A. V., Prykhodko K. S., Makarova L. M., 2022
DOI 10.15588/1607-3274-2022-3-14

АНОТАЦІЯ

Актуальність проблеми оцінювання тривалості розробки програмного забезпечення (ПЗ) на Java для персональних комп'ютерів (ПК) обумовлена наступним чинниками: по-перше, невдале оцінювання тривалості часто є основною причиною невдалої реалізації програмних проектів; по-друге, Java є популярною мовою; і, по-третє, ПК є широко поширеним багатоцільовим комп'ютером. Об'єктом дослідження є процес оцінювання тривалості розробки ПЗ на Java для ПК. Предметом дослідження є моделі нелінійної регресії для оцінювання тривалості розробки ПЗ на Java для ПК.

Мета. Метою роботи є побудова нелінійних регресійних моделей для оцінювання тривалості розробки ПЗ в Java для ПК на основі нормалізуючого перетворення у вигляді десяткового логарифму та видалення викидів у даних для підвищення достовірності оцінювання порівняно з моделлю ISBSG. для платформи ПК.

Метод. За допомогою відповідних методів на основі нормалізуючих перетворень для негаусових даних побудовано моделі, довірчі інтервали та інтервали прогнозування нелінійних регресій для оцінки тривалості розробки ПЗ на Java для ПК. Методи побудови моделей, довірчих інтервалів та інтервалів прогнозування нелінійних регресій базуються на нормалізуючих перетвореннях. Також ми застосовуємо видалення викидів для побудови моделей. Загалом, вищезазначене призводить до зменшення середньої величини відносної похибки, ширини довірчих інтервалів та інтервалів прогнозування порівняно з нелінійними моделями, побудованими без застосування видалення викидів у процесі побудови моделей.

Результати. Проведено порівняння побудованої на основі десяткового логарифму моделі з моделями нелінійної регресії на основі перетворень Джонсона (для сімейства S_B) та Бокса-Кокса як одновимірних, так і двовимірних.

Висновки. Модель нелінійної регресії для оцінювання тривалості розробки ПЗ на Java для ПК побудована на основі перетворення десяткового логарифма. Ця модель, порівняно з іншими моделями нелінійної регресії, має менші значення ширини довірчих інтервалів та інтервалів прогнозування для трудовитрат, які перевищують 900 людино-годин. Перспективи подальших досліджень можуть передбачати застосування двовимірних нормалізуючих перетворень і наборів даних для побудови нелінійних регресійних моделей для оцінювання тривалості розробки ПЗ іншими мовами для ПК та інших платформ, наприклад, мейнфреймів.

КЛЮЧОВІ СЛОВА: тривалість, розробка програмного забезпечення, Java, персональний комп'ютер, нелінійна регресійна модель, нормалізуюче перетворення, неганусові дані, ISBSG.

УДК 004.412:519.237.5

НЕЛИНЕЙНЫЕ РЕГРЕССИОННЫЕ МОДЕЛИ ДЛЯ ОЦЕНИВАНИЯ ПРОДОЛЖЕННОСТИ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ НА JAVA ДЛЯ ПК ПО ДАННЫМ ISBSG 2021 года

Приходько С. Б. – д-р техн. наук, профессор, заведующий кафедрой программного обеспечения автоматизированных систем Национального университету кораблестроения имени адмирала Макарова, Николаев, Украина.

Пухалевич А. В. – канд. техн. наук, преподаватель кафедры программного обеспечения автоматизированных систем Национального университету кораблестроения имени адмирала Макарова, Николаев, Украина.

Приходько Е. С. – канд. техн. наук, доцент кафедры информационных систем и технологий Национального университету кораблестроения имени адмирала Макарова, Николаев, Украина.

Макарова Л. Н. – канд. техн. наук, доцент кафедры программного обеспечения автоматизированных систем Национального университету кораблестроения имени адмирала Макарова, Николаев, Украина.

АННОТАЦИЯ

Актуальность проблемы оценивания продолжительности разработки программного обеспечения (ПО) на Java для персональных компьютеров (ПК) обусловлена следующими факторами: во-первых, неудачное оценивание продолжительности часто является основной причиной неудачной реализации программных проектов; во-вторых, Java является популярным языком; и, в-третьих, ПК является широко распространенным многоцелевым компьютером. Объектом исследования является процесс оценки продолжительности разработки программного обеспечения в Java для ПК. Предметом исследования являются модели нелинейной регрессии для оценки продолжительности разработки ПО на Java для ПК.

Цель. Целью работы является построение нелинейных регрессионных моделей для оценки продолжительности разработки ПО на Java для ПК с использованием нормализующего преобразования в виде десятичного логарифма и удаления выбросов в данных для повышения достоверности оценивания по сравнению с моделью ISBSG для платформы ПК.

Метод. С помощью соответствующих методов на основе нормализующих преобразований для негауссовых данных построена модель, доверительные интервалы и интервалы прогнозирования нелинейной регрессии для оценки продолжительности разработки ПО на Java для ПК. Методы построения моделей, доверительных интервалов и интервалов прогнозирования нелинейных регрессий базируются на нормализирующих преобразованиях. Также мы используем удаление выбросов для построения модели. В целом вышеупомянутое приводит к уменьшению средней величины относительной погрешности, ширины доверительных интервалов и интервалов прогнозирования по сравнению с нелинейными моделями, построенными без применения удаления выбросов в процессе построения модели.

Результаты. Произведено сравнение построенной на основі десятичного логарифма модели с моделями нелинейной регрессии на основе преобразований Джонсона (для семейства S_B) и Бокса-Кокса как одномерных, так и двумерных.

Выводы. Модель нелинейной регрессии для оценивания продолжительности разработки ПО на Java для ПК построена на основе преобразования десятичного логарифма. Эта модель, по сравнению с другими моделями нелинейной регрессии, имеет меньшие значения ширины доверительных интервалов и интервалов прогнозирования для трудозатрат, превышающих 900 человеко-часов. Перспективы дальнейших исследований могут предусматривать применение двумерных нормали-

© Prykhodko S. B., Pukhalevych A. V., Prykhodko K. S., Makarova L. M., 2022
DOI 10.15588/1607-3274-2022-3-14

зирующих преобразований и наборов данных для построения нелинейных регрессионных моделей для оценивания продолжительности разработки ПО на других языках для ПК и других платформ, например мейнфреймов.

КЛЮЧЕВЫЕ СЛОВА: длительность, разработка программного обеспечения, Java, персональный компьютер, нелинейная регрессионная модель, нормализующее преобразование, негауссовы данные, ISBSG.

ЛІТЕРАТУРА / ЛИТЕРАТУРА

1. Boehm B. W. Software engineering economics / B. W. Boehm. – Englewood Cliffs, NJ : Prentice Hall, 1981. – 768 p.
2. Boehm B.W. Software cost estimation with COCOMO II / [B. W. Boehm, C. Abts, A. W. Brown, et al.]. – Upper Saddle River, NJ: Prentice Hall PTR, 2000. – 506 p.
3. Owais M. Effort, duration and cost estimation in agile software development / M. Owais, R. Ramakishore // 2016 Ninth International Conference on Contemporary Computing (IC3), 2016. – P. 1–5, DOI: 10.1109/IC3.2016.7880216.
4. Abran A. Data collection and industry standards: the ISBSG repository / A. Abran // Software Project Estimation: The Fundamentals for Providing High Quality Information to Decision Makers, IEEE, 2015. – P. 161–184, DOI: 10.1002/9781118959312.ch8.
5. Putnam L. H. A general empirical solution to the macrosoftware sizing and estimating problem / L. H. Putnam // IEEE Transactions on Software Engineering. –1978 July. – Vol. 4, No. 2. – P. 345–361.
6. Exploring the relation between effort and duration in software engineering projects / [S. Oligny, P. Bourque, A. Abran, B. Fournier] // Proceedings of the World Computer Congress, Aug. 2000. – P. 175–178.
7. Prykhodko S. B. Developing PC Software Project Duration Model based on Johnson transformation / S. B. Prykhodko, A. V. Pukhalevich // Proceedings of the 12th International Conference Modern Problems of Radio Engineering, Telecommunications and Computer Science TCSET'2014, Lviv-Slavske, Ukraine. – Lviv : Polytechnic National University, 2014. – P. 114–116.
8. Prykhodko S.B. Confidence interval estimation of PC software project duration regression based on Johnson transformation / S. B. Prykhodko, A. V. Pukhalevich // Радіоелектронні і комп'ютерні системи. – Харків, 2014. – № 2 (66) – С. 104–107. – ISSN: 1814-4225
9. López-Martín C. Neural networks for predicting the duration of new software projects / C. López-Martín, A. Abran // Journal of Systems and Software. – 2015, Vol. 101. – P. 127–135. DOI: 10.1016/J.JSS.2014.12.002
10. Pospieszny P. An effective approach for software project effort and duration estimation with machine learning algorithms / P. Pospieszny, B. Czarnacka-Chrobot, A. Kobylinski // Journal of Systems and Software. – 2018. – P. 184–196. DOI: 10.1016/J.JSS.2017.11.066
11. Prykhodko S. Mathematical modeling of non-Gaussian dependent random variables by nonlinear regression models based on the multivariate normalizing transformations / S. Prykhodko, N. Prykhodko // Mathematical Modeling and Simulation of Systems : 15th International Scientific-practical Conference MODS'2020, Chernihiv, Ukraine, June 29 – July 01, 2020 : selected papers. – Springer, Cham., 2021. – P. 166–174. – (Advances in Intelligent Systems and Computing, Vol. 1265). DOI: 10.1007/978-3-030-58124-4_16
12. Bates D. M. Nonlinear regression analysis and its applications / D. M. Bates, D. G. Watts. – New York : John Wiley & Sons, 1988. – 384 p. DOI:10.1002/9780470316757
13. Seber G.A.F. Nonlinear regression / G.A.F. Seber, C. J. Wild. – New York : John Wiley & Sons, 1989. – 768 p. DOI: 10.1002/0471725315
14. Ryan T.P. Modern regression methods / T. P. Ryan. – New York : John Wiley & Sons, 1997. – 529 p. DOI: 10.1002/9780470382806
15. Drapper N. R. Applied regression analysis / N. R. Drapper, H. Smith. – New York : John Wiley & Sons, 1998. – 736 p.
16. Johnson R. A. Applied multivariate statistical analysis / R. A. Johnson, D. W. Wichern. – Pearson Prentice Hall, 2007. – 800 p.
17. Chatterjee S. Handbook of regression analysis / S. Chatterjee, J. S. Simonoff. – New York : John Wiley & Sons, 2013. – 236 p. DOI: 10.1002/9781118532843
18. Detecting Outliers in Multivariate Non-Gaussian Data on the basis of Normalizing Transformations / [S. Prykhodko, N. Prykhodko, L. Makarova, et al.] // Electrical and Computer Engineering : the 2017 IEEE First Ukraine Conference (UKRCON) «Celebrating 25 Years of IEEE Ukraine Section», Kyiv, Ukraine, May 29 – June 2, 2017 : proceedings. – Kyiv : IEEE, 2017. – P. 846–849. DOI: 10.1109/UKRCON.2017.8100366
19. Application of the Squared Mahalanobis Distance for Detecting Outliers in Multivariate Non-Gaussian Data / [S. Prykhodko, N. Prykhodko, L. Makarova, et al.] // Radioelectronics, Telecommunications and Computer Engineering : 14th International Conference on Advanced Trends (TCSET), Lviv-Slavske, Ukraine, February 20–24, 2018 : proceedings. – P. 962–965. DOI: 10.1109/TCSET.2018.8336353
20. Prykhodko S. Estimating the efforts of mobile application development in the planning phase using nonlinear regression analysis / S. Prykhodko, N. Prykhodko, K. Knyrik // Applied Computer Systems. – 2020. – Vol. 25, No. 2. – P. 172–179. DOI: 10.2478/acss-2020-0019
21. Mardia K. V. Measures of multivariate skewness and kurtosis with applications / K. V. Mardia // Biometrika. – 1970. – Vol. 57. – P. 519–530. DOI: 10.1093/biomet/57.3.519
22. Mardia K. V. Applications of some measures of multivariate skewness and kurtosis in testing normality and robustness studies / K. V. Mardia // Sankhya: The Indian Journal of Statistics, Series B (1960–2002). – 1974. – Vol. 36, Issue 2. – P. 115–128.
23. Campbell S. L. Modeling and simulation in Scilab/Scicos / S. L. Campbell, J.-P. Chancelier, R. Nikoukhah. – Springer, 2005. – 313 p.
24. Foss T. A simulation study of the model evaluation criterion MMRE / T. Foss, E. Stensrud, B. Kitchenham, I. Myrtveit // IEEE Transactions on software engineering. – 2003. – Vol. 29, Issue 11. – P. 985–995. DOI: 10.1109/TSE.2003.1245300
25. Port D. Comparative studies of the model evaluation criterions MMRE and PRED in software cost estimation research / D. Port, M. Korte // Empirical Software Engineering and Measurement : the 2nd ACM-IEEE International Symposium ESEM, Kaiserslautern, Germany, October, 2008 : proceedings. – New York : ACM, 2008. – P. 51–60.

© Prykhodko S. B., Pukhalevych A. V., Prykhodko K. S., Makarova L. M., 2022
DOI 10.15588/1607-3274-2022-3-14