Early LOC Estimation of Web Apps Created Using Yii Framework by Nonlinear Regression Models

SERGIY PRYKHODKO, IVAN SHUTKO, ANDRII PRYKHODKO
Department of Software of Automated Systems
Admiral Makarov National University of Shipbuilding
Heroes of Ukraine Ave., 9, Mykolaiv, 54007
UKRAINE

Abstract: - We have performed early LOC estimation of Web applications (apps) created using the Yii framework by three nonlinear regression models with three predictors based on the normalizing transformations. We used two univariate transformations (the decimal logarithm and the Box-Cox transformation) and the Box-Cox four-variate transformation for constructing nonlinear regression models. The nonlinear regression model constructed by the Box-Cox four-variate transformation has better size prediction results compared to other regression ones based on the univariate transformations.

Key-Words: LOC, estimation, Web application, Yii framework, nonlinear regression model, normalizing transformation

Received: May 22, 2021. Revised: October 12, 2021. Accepted: October 29, 2021. Published: November 23, 2021.

1 Introduction

Early software size estimation is one of the project managers' significant problems in evaluating apps development efforts [1-4]. According to [4], "Software size is the major determinant of software project effort." Failed software size estimating is often the main contributor to failed effort estimates and, in consequence, failed projects.

Despite a large number of currently existing various methods and models for estimating the software size [5-15], research in this direction does not stop [16-18]. This is primarily due to the low accuracy of estimating the size of the software in the early stages of its development. One way to solve this problem is to develop appropriate models for estimating the size of the software, which is developed as in a specific programming language [5, 8, 9, 12, 14] and for a specific type of app [7-10, 14, 15].

Lines of code (LOC) and function points (FPs) are most commonly used as measures of size in existing software effort estimation methods and models. As known [4], both of these metrics have their advantages and disadvantages when used for software effort estimation. Although the FPs-based measure has the advantage over the LOC in that it does not depend on the technologies used – in particular, the programming language, however, the assessment of efforts requires taking into account such factors (environmental factors). Taking into account the above factors can be ensured by appropriate models for estimating the LOC-based measure.

Today many Web apps are created using PHP frameworks making app development faster. Yii is a fast, secure, and efficient PHP framework (https://www.yiiframework.com/). However, there are no regression models for estimating the software size of Web apps created using the Yii framework. There are some regression equations, both linear [8, 9] and nonlinear [14] ones, for estimating the software size of information open-source PHP-based systems. Only in [19], a nonlinear regression model to estimate the software size of Web apps created using the Laravel framework was built. This demands the construction of the models for early software size estimation of Web apps created using the Yii framework.

Although machine learning methods are becoming increasingly popular for software size estimation [17, 18], methods based on nonlinear regression analysis have not yet reached their full potential [20, 21]. We suggest using the nonlinear regression models for estimating the size of Web apps created using the Yii framework because, firstly, there are two random variables, both dependent variable (response) and an error term, in a regression model, and, secondly, the size (response) distribution is not Gaussian. We apply the technique for constructing nonlinear regression

models based on the multivariate normalizing transformations and prediction intervals [21]. In this technique, prediction intervals of nonlinear regressions are used to detect the outliers in constructing a nonlinear regression model. Usually, the above process is iterative since we repeat building the model for new data after the outlier cutoff. If there are no outliers, the process of constructing the nonlinear regression model ends.

2 Problem Formulation

Suppose given the original sample as the four-dimensional non-Gaussian data set: actual software size in the thousand lines of code (KLOC) *Y*, the total number of classes X_1, the average number of methods per class X_2, the average of Depth of Inheritance Tree (DIT) per class X_3 in a class diagram from *N* Web apps. Suppose that there are a bijective four-variate normalizing transformation of non-Gaussian random vector $\mathbf{P}=\{Y,X_1,X_2,X_3\}^T$ to Gaussian random vector $\mathbf{T}=\{Z_Y,Z_1,Z_2,Z_3\}^T$ is given by

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

and the inverse transformation for (1)

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

$\boldsymbol{\psi}$ is a vector of normalizing transformation (1), $\boldsymbol{\psi}=\{\psi_Y,\psi_1,\psi_2,\psi_3\}^T$.

It is required to build the nonlinear regression model in the form $Y=Y(X_1,X_2,X_3,\varepsilon)$ based on the transformations (1) and (2) to estimate the software size (in KLOC) of Web apps created using the Yii framework. Here ε is the error term that is the Gaussian random variable to describe residuals, $\varepsilon\sim N(0,\sigma_\varepsilon^2)$, σ_ε is the standard deviation.

3 Problem Solution

To build a nonlinear regression model for estimating the size of Web apps created using the Yii framework, we collected data from 40 apps hosted on GitHub (https://github.com). We obtained the data set by the PhpMetrics tool (https://phpmetrics.org/) around the following variables: actual software size (in KLOC) *Y*, the total number of classes X_1, the average number of methods per class X_2, and the DIT average per class X_3. Table 1 contains that data set. We chose the above predictors X_1, X_2, and X_3 for two reasons. Firstly, these predictors can be obtained from the class diagram, and, secondly, there is no multicollinearity between these predictors according to [22, 23] since variance inflation factors (VIFs) for predictors X_1, X_2, and X_3 are equal to 1.44, 1.22, and 1.64, respectively.

Table 1. The data set and SMD values

No	*Y*	X_1	X_2	X_3	SMD	SMD_Z
1	1.333	30	5.57	2.10	3.07	4.71
2	42.543	401	7.01	1.28	5.57	4.86
3	55.471	598	7.66	1.27	6.75	6.52
4	1.296	33	3.97	1.89	0.31	0.44
5	10.175	174	5.16	1.55	2.65	1.10
6	1.374	31	4.45	1.93	0.47	0.85
7	1.003	25	4.28	2.00	0.61	1.15
8	4.496	132	3.76	1.96	0.27	1.16
9	44.998	1149	3.71	1.90	13.37	7.67
10	4.587	194	2.63	1.65	3.42	3.89
11	0.213	12	1.92	2.00	2.69	4.84
12	10.389	76	6.43	1.88	3.06	11.24
13	3.321	102	3.30	1.68	1.93	1.03
14	2.525	53	4.81	2.40	7.67	7.92
15	29.477	665	4.82	1.80	3.14	3.61
16	0.805	20	4.45	1.78	0.91	2.66
17	6.114	97	5.35	1.90	1.12	1.30
18	3.312	96	4.00	2.11	1.32	2.24
19	0.883	32	3.31	2.11	1.39	1.72
20	47.417	1286	3.71	1.61	**18.01**	6.29
21	3.103	61	4.62	2.07	1.35	1.72
22	36.615	373	7.49	1.37	5.40	4.96
23	3.731	77	3.25	1.57	3.56	4.50
24	42.963	416	7.69	1.45	5.80	4.79
25	0.341	18	2.22	2.20	3.62	4.28
26	0.211	10	2.10	1.67	4.42	7.19
27	1.053	25	4.92	1.82	1.01	2.81
28	10.799	214	3.52	2.04	1.07	6.06
29	0.27	15	0.87	2.00	5.63	6.45
30	7.579	150	4.57	2.00	0.76	1.70
31	0.12	4	1.50	2.00	3.80	7.87
32	1.008	29	4.45	2.17	2.33	3.31
33	0.51	12	3.75	1.88	0.43	2.63
34	0.968	25	3.60	2.09	1.15	1.27
35	0.791	16	2.81	2.00	1.20	7.42
36	119.314	1428	4.26	1.06	**30.87**	11.12
37	1.691	57	3.32	1.73	1.48	1.11
38	2.431	74	3.51	1.58	3.38	1.77
39	2.984	80	4.18	1.64	2.08	1.08
40	27.627	603	4.93	1.59	2.96	2.76

We checked the four-dimensional data from Table 1 for multivariate outliers. This is step 1 according to [21]. But before that, we tested the normality of multivariate data from Table 1 because well-known statistical methods (for example, multivariate outlier detection based on the squared Mahalanobis distance (SMD)) are applied to detect outliers in multivariate data under the assumption that the data is described by a Gaussian distribution

[24, 25]. We used a multivariate normality test proposed by Mardia [26, 27]. This test is based on measures of multivariate skewness β_1 and kurtosis β_2.

According to the Mardia test, the distribution of four-dimensional data from Table I is not Gaussian since the test statistic for multivariate skewness $N\beta_1/6$ of this data, which equals 197.98, is greater than the quantile of the Chi-Square distribution, which is 40.00 for 20 degrees of freedom and 0.005 significance level. Analogically, the test statistic for multivariate kurtosis β_2, which equals 46.24, is greater than the value of the Gaussian distribution quantile, which is 29.64 for the mean of 24, the variance of 4.8, and 0.005 significance level. Because we used the statistical technique [25] to detect multivariate outliers in the four-dimensional non-Gaussian data from Table 1 based on the multivariate normalizing transformations and the SMD for normalized data. To normalize the data from Table 1, we applied the four-variate Box-Cox transformation with components [24].

$$Z_j = x(\lambda_j) = \begin{cases} \left(X_j^{\lambda_j} - 1\right)/\lambda_j, & \text{if} \quad \lambda_j \neq 0; \\ \ln(X_j), & \text{if} \quad \lambda_j = 0. \end{cases} \tag{3}$$

Here Z_j is a Gaussian variable; λ_j is a parameter of the Box-Cox transformation, $j = 1,2,3$. The variable Z_Y is defined analogously (3) with the only difference that instead of Z_j, X_j, and λ_j should be put respectively Z_Y, Y, and λ_Y.

The parameter estimates of the four-variate Box-Cox transformation for the data from Table 1 are calculated by the maximum likelihood method according to [24] and are $\hat{\lambda}_Y = -0.039927$, $\hat{\lambda}_1 = -0.015134$, $\hat{\lambda}_2 = 0.709637$, $\hat{\lambda}_3 = 1.604595$.

Table 1 contains the SMD for normalized data (SMD_Z), which is transformed using the four-variate Box-Cox transformation. The SMD_Z values from Table 1 indicate there is no multivariate outlier in four-dimensional non-Gaussian data since the SMD_Z values for all data rows are less than the quantile of the Chi-Square distribution, which equals 14.86 for 4 degrees of freedom and 0.005 significance level. Note, for data without normalization, row 36 is the multivariate outlier since the SMD value for row 36 is greater than the above quantile. In Table 1, this SMD value is highlighted in bold.

The nonlinear regression model with three predictors for estimating the size of Web apps created using the Yii framework is built based on the four-variate Box-Cox transformation for 40 data rows from Table 1 according to [21] and has the form

$$Y = \left[\hat{\lambda}_Y\left(\hat{Z}_Y + \varepsilon\right) + 1\right]^{1/\hat{\lambda}_Y}, \tag{4}$$

where ε is a Gaussian random variable, $\varepsilon \sim N\left(0, \sigma_\varepsilon^2\right)$, with the estimate $\hat{\sigma}_\varepsilon$ of 0.21317; $\hat{Z}_Y$ is a prediction result by the linear regression equation $\hat{Z}_Y = \hat{b}_0 + \hat{b}_1 Z_1 + \hat{b}_2 Z_2 + \hat{b}_3 Z_3$ for normalized data, which are transformed by the four-variate Box-Cox transformation with components (3); $\hat{b}_0 = -3.95701$, $\hat{b}_1 = 1.02178$, $\hat{b}_2 = 0.36521$, $\hat{b}_3 = -0.10204$.

According to [21], after constructing a model (4), we have to find the nonlinear regression prediction interval

$$\psi_Y^{-1}\left(\hat{Z}_Y \pm t_{\alpha/2,\nu} S_{Z_Y}\left\{1 + \frac{1}{N} + \left(\mathbf{z}_X^+\right)^T \mathbf{S}_Z^{-1}\left(\mathbf{z}_X^+\right)\right\}^{1/2}\right), \tag{5}$$

where ψ_Y is the transformation (3) for Y, $\psi_Y^{-1} = \left(\hat{\lambda}_Y Z_Y + 1\right)^{1/\hat{\lambda}_Y}$; $t_{\alpha/2,\nu}$ is a student's t-distribution quantile with $\alpha/2$ significance level and ν degrees of freedom; $\nu = N - k - 1$; k is a number of independent variables (in our case, k is 3); $\mathbf{z}_X^+$ is a vector with components $Z_{1_i} - \overline{Z}_1$, $Z_{2_i} - \overline{Z}_2$, $Z_{3_i} - \overline{Z}_3$ for i-row; $\overline{Z}_j = \frac{1}{N}\sum_{i=1}^{N} Z_{j_i}$, $j = 1,2,3$; $S_{Z_Y}^2 = \frac{1}{\nu}\sum_{i=1}^{N}\left(Z_{Y_i} - \hat{Z}_{Y_i}\right)^2$, $\nu = N - k - 1$; $\mathbf{S}_Z$ is a 3×3 matrix

$$\mathbf{S}_Z = \begin{pmatrix} S_{Z_1Z_1} & S_{Z_1Z_2} & S_{Z_1Z_3} \\ S_{Z_1Z_2} & S_{Z_2Z_2} & S_{Z_2Z_3} \\ S_{Z_1Z_3} & S_{Z_2Z_3} & S_{Z_3Z_3} \end{pmatrix}. \tag{6}$$

In (6) $S_{Z_qZ_r} = \sum_{i=1}^{N}\left[Z_{q_i} - \overline{Z}_q\right]\left[Z_{r_i} - \overline{Z}_r\right]$, $q, r = 1,2,3$.

For the data normalized by the four-variate Box-Cox transformation from 40 Web apps, the matrix (6) is the following:

$$\mathbf{S}_Z = \begin{pmatrix} 74.488 & 30.684 & -12.584 \\ 30.684 & 42.674 & -6.364 \\ -12.584 & -6.364 & 6.222 \end{pmatrix}.$$

Table 2 contains the values of lower (LB) and upper (UB) bounds of the nonlinear regression prediction interval calculated by (5) based on the four-variate Box-Cox transformation for 0.05 significance level in the first iteration for 40 data rows from Table 1. In Table 2, we denoted LB and UB in the first iteration as LB_1 and UB_1, respectively.

Table 2. LB and UB of nonlinear regression prediction intervals in various iterations

No	Y	The first iteration		The second iteration		
		LB_1	UB_1	SMD_Z	LB_2	UB_2
1	1.333	1.037	2.735	4.48	1.050	2.277
2	42.543	22.326	67.979	5.17	23.897	58.294
3	55.471	39.633	125.380	6.54	42.442	106.842
4	1.296	0.824	2.081	0.54	0.863	1.805
5	10.175	5.953	16.366	1.14	6.368	14.273
6	1.374	0.863	2.193	1.07	0.894	1.881
7	1.003	0.667	1.682	1.20	0.689	1.442
8	4.496	2.985	7.975	1.07	3.211	7.033
9	44.998	25.904	82.363	7.45	29.891	75.611
10	4.587	3.323	9.153	4.29	3.724	8.372
11	0.213	0.182	0.446	5.61	0.195	0.397
12	10.389	3.266	8.957	-	-	-
13	3.321	2.140	5.641	0.92	2.330	5.047
14	2.525	1.411	3.947	7.84	1.460	3.304
15	29.477	20.673	61.423	3.80	22.913	54.788
16	0.805	0.575	1.462	2.48	0.596	1.253
17	6.114	3.284	8.797	2.22	3.413	7.497
18	3.312	2.248	6.010	2.04	2.385	5.223
19	0.883	0.655	1.651	1.62	0.689	1.440
20	47.417	30.874	96.950	6.59	36.096	90.409
21	3.103	1.694	4.443	2.55	1.762	3.802
22	36.615	22.822	69.684	4.85	24.061	58.802
23	3.731	1.621	4.270	6.17	1.764	3.820
24	42.963	26.550	81.476	4.71	27.866	68.382
25	0.341	0.279	0.695	4.55	0.297	0.614
26	0.211	0.167	0.420	7.50	0.180	0.375
27	1.053	0.790	2.028	2.63	0.813	1.723
28	10.799	4.417	12.309	7.83	4.826	10.937
29	0.27	0.158	0.399	6.81	0.172	0.362
30	7.579	4.128	11.230	2.14	4.383	9.745
31	0.12	0.059	0.142	9.65	0.063	0.127
32	1.008	0.774	1.991	3.09	0.797	1.693
33	0.51	0.297	0.737	3.22	0.309	0.637
34	0.968	0.558	1.399	1.83	0.583	1.212
35	0.791	0.304	0.747	-	-	-
36	119.314	42.952	140.755	11.64	51.178	133.360
37	1.691	1.213	3.117	1.08	1.303	2.764
38	2.431	1.670	4.387	1.67	1.807	3.902
39	2.984	2.129	5.588	0.99	2.272	4.904
40	27.627	19.976	58.505	3.03	22.184	52.394

As we observe in Table 2, there are two values of Y for Web apps 12 and 35 that are out of the prediction intervals computed by (5) for a significance level of 0.05. Next, we erased data rows 12 and 35. The first iteration is completed. And we go to step 1 of the second iteration according to [21].

We checked the four-dimensional data from Table 1 (without rows 12 and 35) for multivariate outliers. To do this, we normalized 38 data rows using the four-variate Box-Cox transformation with components, which are defined by (3). The parameter estimates of the four-variate Box-Cox transformation for 38 data rows from Table 1 (without data rows 12 and 35) are calculated by the maximum likelihood method according to [24] and are $\hat{\lambda}_Y = -0.040881$, $\hat{\lambda}_1 = -0.012110$, $\hat{\lambda}_2 = 0.684253$, $\hat{\lambda}_3 = 1.336337$.

Before outlier detection in the second iteration, we checked the multivariate normality of 38 rows of normalized data from Table 1 (without data rows 12 and 35) by a test proposed by Mardia [26].

According to Mardia's test, the distribution of 38 rows of normalized data from Table I (excluding data rows 12 and 35) is Gaussian since the test statistic for multivariate skewness $N\beta_1/6$ of this data, which equals 22.11, is less than the quantile of the Chi-Square distribution, which is 40.00 for 20 degrees of freedom and 0.005 significance level. Analogically, the test statistic for multivariate kurtosis β_2, which equals 23.44, is less than the quantile of the Gaussian distribution, which is 29.79 for the mean of 24, the variance of 5.053, and 0.005 significance level. Because we used the statistical technique [25] to detect multivariate outliers in the four-dimensional non-Gaussian data from Table 1 (without data rows 12 and 35) based on the multivariate normalizing transformations and the SMD for normalized data.

The SMD_Z values from Table 2 indicate there is no multivariate outlier in four-dimensional non-Gaussian data from Table 1 (without rows 12 and 35) since the SMD_Z values for 38 data rows are less than the quantile of the Chi-Square distribution, which equals 14.86 for 4 degrees of freedom and 0.005 significance level.

Next, we built model (4) based on the four-variate Box-Cox transformation for 38 data rows. In this case, the parameters estimates of the model (4) are the following: $\hat{b}_0 = -3.94744$, $\hat{b}_1 = 1.02437$, $\hat{b}_2 = 0.35020$, $\hat{b}_3 = -0.14416$, $\hat{\sigma}_\varepsilon = 0.16918$.

After constructing a model (4), we calculated the nonlinear regression prediction interval for 38 data rows in the second iteration (see Table 2). For the data normalized by the four-variate Box-Cox transformation from 38 Web apps, the matrix (6) is the following:

$$\mathbf{S}_Z = \begin{pmatrix} 74.222 & 28.777 & -10.760 \\ 28.777 & 37.154 & -5.262 \\ -10.760 & -5.262 & 4.563 \end{pmatrix}.$$

In Table 2, we denoted LB and UB in the second iteration as LB_2 and UB_2, respectively. We highlighted the row numbers with the data outliers in bold, and a dash (-) shows the exception of the relevant numbers of data at the second iteration. The LB_2 and UB_2 values indicate there are no values of Y for 38 data rows that are not out of the prediction intervals computed by (5) for a significance level of 0.05. Because we completed the stages' iterations and constructed a nonlinear regression model (4) with 38 Web apps data.

Also, to estimate the size of Web apps created using the Yii framework, we built two nonlinear regression models with three predictors based on the univariate normalizing transformations (the decimal logarithm, and the Box-Cox transformation) for the same 38 Web apps data.

The nonlinear regression model with three predictors based on the univariate Box-Cox transformation has the form (4) too, but with the only difference that parameters estimates are the following: $\hat{\lambda}_Y = -0.054599$, $\hat{\lambda}_1 = -0.080356$, $\hat{\lambda}_2 = 0.714786$, $\hat{\lambda}_3 = 2.139290$, $\hat{b}_0 = -4.52586$, $\hat{b}_1 = 1.37338$, $\hat{b}_2 = 0.30859$, $\hat{b}_3 = -0.10751$, $\hat{\sigma}_\varepsilon = 0.17912$.

For the data normalized by the univariate Box-Cox transformation from 38 Web apps, the matrix (6) is the following:

$$\mathbf{S}_Z = \begin{pmatrix} 40.258 & 22.636 & -11.882 \\ 22.636 & 40.334 & -8.213 \\ -11.882 & -8.213 & 11.332 \end{pmatrix}.$$

The nonlinear regression model based on the decimal logarithm transformation has the form

$$Y = 10^{\varepsilon+\hat{b}_0} X_1^{\hat{b}_1} X_2^{\hat{b}_2} X_3^{\hat{b}_3}, \qquad (7)$$

where the estimators for parameters are: $\hat{b}_0 = -1.63446$, $\hat{b}_1 = 1.00735$, $\hat{b}_2 = 0.743069$, $\hat{b}_3 = -0.78570$. The estimate $\hat{\sigma}_\varepsilon$ is 0.089021.

For the data normalized by the decimal logarithm transformation from 38 Web apps, the matrix (6) is the following:

$$\mathbf{S}_Z = \begin{pmatrix} 15.638 & 2.477 & -1.101 \\ 2.477 & 1.317 & -0.194 \\ -1.101 & -0.194 & 0.211 \end{pmatrix}.$$

To evaluate the prediction accuracy of the nonlinear regression models we applied the standard metrics R^2, MMRE, and PRED(0.25). MMRE and PRED(0.25) are accepted as standard evaluations of prediction results by regression models. These metrics are applied in software engineering too [28, 29]. 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 3 for models (4) for both the univariate and four-variate Box-Cox transformations, and model (7).

Table 3. The prediction accuracy metrics of the nonlinear regression models

Metrics	univariate		bivariate
	Log10	Box-Cox	Box-Cox
R^2	0.9656	0.8925	0.9249
MMR_{min}	0.0018	0.0018	0.0087
MMR_{max}	0.4119	0.4000	0.3295
MMRE	0.1705	0.1463	0.1439
PRED(0.25)	0.7632	0.8421	0.8684

The values of these metrics are acceptable for all models. These values indicate good prediction accuracy of the nonlinear regression models (4) and (7) for estimating the size of Web apps created using the Yii framework. However, model (4) based on the four-variate Box-Cox transformation has the best MMRE and PRED(0.25) values.

Also, Table 3 contains minimum and maximum values of MRE denoted MMR_{min} and MMR_{max}, respectively. As we observe in Table 3, we have the smallest MMR_{max} value for model (4) based on the four-variate Box-Cox transformation. The above indicates the advantages of using model (4) based on the four-variate Box-Cox transformation for estimating the size of Web apps created using the Yii framework.

The advantage of using model (4) based on the four-variate Box-Cox transformation in comparison to other constructed models based on univariate

transformations is also indicated by the width of the confidence and prediction intervals. We calculated the confidence intervals of nonlinear regressions by (5) with the only difference that in the sum in curly brackets, there is not 1.

The widths of the confidence interval of nonlinear regression based on the Box-Cox four-variate transformation are less than for nonlinear regression based on the Box-Cox univariate transformation for 34 (with the difference up to 21%) from 38 rows of data (except rows 9, 20, 31, and 36 with the difference up to 11%). Also, the widths of the confidence interval of nonlinear regression based on the Box-Cox four-variate transformation are less than for nonlinear regression based on the decimal logarithm univariate transformation for 33 (with the difference up to 35%) from 38 rows of data (except rows 2, 3, 14, 22, and 24 with the difference up to 31%).

Approximately the same results are obtained for the prediction intervals of nonlinear regressions. The widths of the prediction interval of nonlinear regression based on the Box-Cox four-variate transformation are less than for nonlinear regression based on the Box-Cox univariate transformation for 33 (with the difference up to 18%) from 38 data rows (except rows 9, 20, 31, 33, and 36 with the difference up to 11%). Also, the widths of the prediction interval of nonlinear regression based on the Box-Cox four-variate transformation are less than for nonlinear regression based on the decimal logarithm univariate transformation for 32 (with the difference up to 40%) from 38 data rows (except rows 2, 3, 9, 15, 22, and 24 with the difference up to 20%).

4 Discussion

The four-variate distribution of the data from Table 1 is not Gaussian what the Mardia test for multivariate normality based on measures of the multivariate skewness and kurtosis indicates. Because we use the statistical technique [25] to detect multivariate outliers in the four-dimensional non-Gaussian data from Table I based on the multivariate normalizing transformations and the SMD for normalized data. According to [25], there are no multivariate outliers in four-dimensional non-Gaussian data from Table I based on the Box-Cox four-variate transformation. Note, we have the four-variate outlier in the data from Table 1 (data row 18) without applying normalization. This may be explained by the four-variate distribution of the data from Table 1 is not Gaussian. Also, we have the four-variate outlier for the data from Table 1 (data row 29) based on the univariate transformation in the decimal logarithm form. This may be explained by the poor normalization of four-dimensional non-Gaussian data from Table 1 using the univariate transformation in the decimal logarithm form.

We apply the four-variate Box-Cox normalizing transformation to build the nonlinear regression model for estimating the size of Web apps created using the Yii framework based on the appropriate technique [21] since there are outliers in the data from Table 1, which are detected in the model construction process by the nonlinear regression prediction interval (see Table 2).

Note, that in our case, the data normalization using the univariate transformations, both the decimal logarithm and Box-Cox ones, leads 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 four-variate transformation. Also, the MMRE value is smaller, and PRED(0.25) value is bigger for the model (4) for the Box-Cox four-variate transformation in comparison with all other nonlinear models based on the univariate transformations. This may be explained by the best four-variate normalization of non-Gaussian data from Table 1 using the Box-Cox four-variate transformation that takes into account the correlation between the variables.

The obtained results and results from [19] indicate that constructing a nonlinear regression model to estimate the size (in KLOC) of Web apps created using the specific framework (Yii in our case and Laravel in [19]) by a technique [21] leads an increase of estimation confidence.

5 Conclusion

Early LOC estimation (in KLOC) of Web apps created using the Yii framework by nonlinear regression models with three predictors based on the normalizing transformations, both univariate and multivariate ones, is performed. The nonlinear regression model constructed using the four-variate Box-Cox transformation has better size prediction results compared to other regression ones based on the univariate transformations (the decimal logarithm and Box-Cox).

To construct nonlinear regression models with multiple predictors for estimating the software size, it needs to apply multivariate normalizing transformations and outlier detection.

References:

[1] B.W. Boehm et al., *Software cost estimation with COCOMO II*, Prentice-Hall, Englewood Cliffs, NJ, 2000.

[2] M. Ruhe, R. Jeffery, and I. Wieczorek, Cost estimation for Web applications, *Proceedings of the International Conference on Software Engineering*, 2003, pp. 285–294.

[3] M. Jorgensen and M. Shepperd, A systematic review of software development cost estimation studies, *IEEE Transactions on Software Engineering*, Vol. 33, No. 1, 2007, pp. 33-53.

[4] A. Trendowicz and R. Jeffery. *Software project effort estimation. foundations and best practice guidelines for success*, Springer International Publishing, 2014. DOI: https://doi.org/10.1007/978-3-319-03629-8.

[5] J. Kaczmarek and M. Kucharski, Size and effort estimation for applications written in Java, *Information and Software Technology,* 46 (9), 2004, pp. 589-601. DOI: https://doi.org/10.1016/j.infsof.2003.11.001.

[6] L.M. Laird and M.C. Brennan, *Software measurement and estimation. A practical approach. quantitative software engineering series*, Wiley-IEEE Computer Society Press, 2006.

[7] E. Mendes, N. Mosley, and S. Counsell, Web effort estimation, In *Web Engineering*, Emilia Mendes and Nile Mosley (Eds.). Springer, 2006, pp. 29-73.

[8] H.B.K. Tan, Y. Zhao, and H. Zhang, Estimating LOC for information systems from their conceptual data models, *Proceedings of the 28th International Conference on Software Engineering (ICSE '06)*, Shanghai, China, May 20-28, 2006, pp. 321-330. DOI: https://doi.org/10.1145/1134285.1134331

[9] H.B.K. Tan, Y. Zhao, and H. Zhang, Conceptual data model-based software size estimation for information systems, *Transactions on Software Engineering and Methodology*. Vol. 19, Issue 2, October 2009, Article No. 4. DOI: https://doi.org/10.1145/1571629.1571630

[10] K. Lind, R. Heldal, T. Harutyunyan, and T. Heimdahl. CompSize: Automated size estimation of embedded software components, *Proceedings from Joint Conference of the 21st International Workshop on Software Measurement and the 6th International Conference on Software Process and Product Measurement*. Nara, Japan, 2011, pp. 86-95. DOI: https://doi.org/10.1109/IWSM-MENSURA.2011.49

[11] Y. Zifen, An improved software size estimation method based on object-oriented approach. *Proceedings from EEESYM'12: IEEE Symposium on Electrical & Electronics Engineering*. Kuala Lumpur, Malaysia, 2012, pp. 615-617. DOI: https://doi.org/10.1109/EEESym.2012.6258733

[12] M. Kiewkanya and S. Surak, Constructing C++ software size estimation model from class diagram. *Proceedings from CSSE'16: Computer Science and Software Engineering: 13th International Joint Conference*. Khon Kaen, Thailand, 2016, pp. 1-6. DOI: https://doi.org/10.1109/JCSSE.2016.7748880.

[13] R.S. Dewi Sholiq and A.P. Subriadi, A comparative study of software development size estimation method: UCPabc vs Function Points. *Procedia Computer Science*, Vol. 124, 2017, pp. 470-477. DOI: https://doi.org/10.1016/j.procs.2017.12.179.

[14] S. Prykhodko, N. Prykhodko, and L. Makarova, Estimating the software size of open-source PHP-based systems using non-linear regression analysis, *Proceedings of International Conference Advanced Computer Information Technologies (ACIT-2018)*. CEUR Workshop Proceedings, Vol. 2300, 2019, CEUR-WS.org, pp. 199-202,

[15] V.R.N. Neyveli, S.S Sivakumar, D. Arunagiri, C. Arumugam, and A.M. Veeramani, An approach to estimate the size of Web application using IFML User interface model. *Proceedings from AICAI'19: Amity International Conference on Artificial Intelligence*. Dubai, United Arab Emirates, 2019, pp. 292-295. DOI: https://doi.org/10.1109/AICAI.2019.8701268

[16] M. Daud and A.A. Malik, Improving the accuracy of early software size estimation using analysis-to-design adjustment factors (ADAFs), *IEEE Access*, Vol. 9, 2021, pp. 81986-81999, DOI: 10.1109/ACCESS.2021.3085752.

[17] Manisha and R. Rishi, Early size estimation using machine learning, *2021 8th International Conference on Computing for Sustainable Global Development (INDIACom)*, 2021, pp. 757-762, DOI: 10.1109/INDIACom51348.2021.00135.

[18] K. Zhang, X. Wang, J. Ren, and C. Liu, Efficiency improvement of function point-based software size estimation with deep learning model, *IEEE Access*, Vol. 9, 2021, pp. 107124-107136, DOI: 10.1109/ACCESS.2020.2998581.

[19] S.B. Prykhodko, N.V. Prykhodko, M.V. Vorona, and I.A. Belovol, Nonlinear regression model for estimating the size of Web applications created using the Laravel

framework, *Information technology and computer engineering*, Vol. 50, No. 1, 2021, pp. 115-121. [Published in Ukrainian] DOI: https://doi.org/10.31649/1999-9941-2021-50-1-115-121

[20] A.B. Nassif, M. AbuTalib, and L.F. Capretz, Software effort estimation from Use Case diagrams using nonlinear regression analysis. *Proceedings from CCECE'20: IEEE Canadian Conference on Electrical and Computer Engineering.* London, ON, Canada, 2020, pp. 1-4. DOI: https://doi.org/10.1109/CCECE47787.2020.9255712

[21] S. Prykhodko and N. Prykhodko, Mathematical modeling of non-Gaussian dependent random variables by nonlinear regression models based on the multivariate normalizing transformations, *Proceedings from MODS'2020: Mathematical Modeling and Simulation of Systems. Advances in Intelligent Systems and Computing*, Vol. 1265, Springer, Cham, 2021, pp. 166-174. DOI: https://doi.org/10.1007/978-3-030-58124-4_16

[22] D.A. Belsley, E. Kuh, and R.E. Welsch, *Regression diagnostics: Identifying influential data and sources of collinearity*, New York: John Wiley, 1980. DOI: https://doi.org/10.1002/0471725153

[23] S. Chatterjee and B. Price, *Regression analysis by example*, New York: John Wiley & Son, 2012.

[24] R.A. Johnson and D.W. Wichern, *Applied multivariate statistical analysis*, Pearson Prentice Hall, 2007.

[25] S. Prykhodko, N. Prykhodko, L. Makarova, and K. Pugachenko, Detecting outliers in multivariate non-Gaussian data on the basis of normalizing transformations, *Proceedings of the First Ukraine Conference on Electrical and Computer Engineering (UKRCON)*. IEEE, Kyiv, 2017, pp, 846-849. DOI: https://doi.org/10.1109/UKRCON.2017.8100366

[26] K.V. Mardia, Measures of multivariate skewness and kurtosis with applications, *Biometrika*, 57, 1970, pp. 519–530. DOI: https://doi.org/10.1093/biomet/57.3.519

[27] K.V. Mardia, 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), Vol. 36, Issue 2, 1974, pp. 115–128.

[28] T. Foss, E. Stensrud, B. Kitchenham, and I. Myrtveit, A simulation study of the model evaluation criterion MMRE, *IEEE Transactions on software engineering*, Vol. 29, Issue 11, 2003, pp. 985–995. DOI: 10.1109/TSE.2003.1245300

[29] D. Port and M. Korte, Comparative studies of the model evaluation criterions MMRE and PRED in software cost estimation research, *Proceedings of the 2nd ACM-IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*, Kaiserslautern, Germany, October 2008, New York: ACM, 2008, pp. 51–60.

Contribution of individual authors to the creation of a scientific article (ghostwriting policy)

Dr. Sergiy Prykhodko advised on the process of formulating the problem, constructing the models, and analyzing the data.
Ivan Shutko constructed the model based on the four-variate transformation and carried out the analysis.
Andrii Prykhodko collected the data, constructed the models based on the univariate transformations, and carried out the analysis.

Follow: www.wseas.org/multimedia/contributor-role-instruction.pdf

Creative Commons Attribution License 4.0 (Attribution 4.0 International , CC BY 4.0)