УДК 005.8:004.94

SOFTWARE APPLICATION FOR BUILDING A REGRESSION MODEL OF ESTIMATING THE PHP WEB APPLICATIONS EFFORT

Ponomarenko T., candidate of technical science, PhD[1]; Gelenko Y.[2]; Kotula V.[3]
[1,2,3] National University of shipbuilding named by admiral Makarov
[1,2,3] Ukraine, Mykolaiv
[1] tetyana.ponomarenko@nuos.edu.ua; [2] urahelenko@gmail.com; [3] south-agrosnab@ukr.net

Abstract. *The article presents a description of software for building a multifactor regression model to estimate the effort of developing web applications* in PHP.
Keywords: *multifactor regression model; web applications in PHP; development complexity.*

Introduction. Regression analysis is an incredibly powerful machine learning tool used for analyzing data. It is a way of predicting future happenings between a dependent (target) and one or more independent variables (predictor). Regression differs from classification models because it estimates a numerical value, whereas classification models identify which category an observation belongs to. The main uses of regression analysis are forecasting, time series modeling and finding the cause and effect relationship between variables. It is well known that model will be more accurate if we will take into account the specificity of the data to construct the predictive models. As we see nowdays, field of php projects of github is huge - about 750 thousand projects, but each framework, for example Yii1 (420 projects) and Yii3 (223 projects), have its own circumstances, so models which will take into account them will bring more profit in future by the opinion of the authors. In this case datasets should be separated for each of the frameworks. This article deals with issues related with regression models and software creation for its implementation [1, 2, 3].

The aim of the work is to create software for building a regression model to estimate PHP-based projects effort, that it is possible to increase the reliability of the assessment of the dependent variable regression compared to using single-dimensional normalization transformations.

Background. Estimation of unknown parameters of the model is carried out solely based on the values metrics of the observations which we have got from GitHub PHP projects. Every tool from the list [4, 5] could be got for the observation extraction. All values are saved in a file that will be uploaded as in-data for the model construction. In Python for this case we should use Pandas, Matplotlib, Statsmodels, Numpy and Sklearn libraries. Lets import them:

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
import statsmodels.api as sm
from statsmodels.stats.outliers_influence import variance_inflation_factor
from sklearn.linear_model import LinearRegression
```

Multicollinearity occurs when there are two or more independent variables in a multiple regression model, which have a high correlation among themselves. When some features are highly correlated, we might have difficulty in distinguishing between their individual effects on the dependent variable. Multicollinearity can be detected using various techniques, one such technique being the Variance Inflation Factor(VIF). We will import VIF function from statsmodels.stats.outlier. Generally, a VIF above 5 indicates a high multicollinearity [2].

Then we should read the data from the file with observations that we have got from GitHub.

```
from google.colab import files
data = files.upload ()
plt.scatter(data.bugfixsize)
plt.figure(figsize=(10, 6))
plt.hist(data['effort'], bins=50, es='black', color='#2196f3')
plt.xlabel('bugfixsize')
plt.ylabel('pages_of_documentation')
plt.show()
regr = LinearRegression()
regr.fit(X_train, y_train)
print(' r-squared:', regr.score (X_train,y_train))
print('Intersept:', regr.intersept_)
pd.DataFrame(data=regr.coef_,index=X_train.columns, columns = 'coef')
pd.DaraFrame({'coef_name':X_incl_const.colums,'vif':np.around(vif, 2)})
vif = [variance_inflation_factor(exog=X_incl_const.values,exog_inx=1) for i in range (X_incl_const.shape[1])]
```

The non-linear regression model for estimating the effort of PHP projects which we have got from that software is Y=2.341231+1.6159 x1 +0.01438x2 , where x1-size in bugfix in bugs, x2 - size of documentation in pages. R-square 0.96.

Conclusions. Software of the model, built on the basis of multi-dimensional Johnson's family SB normalization transformation, compared to other regressive the models have a higher percentage of projected results, lower average values relative error and range of the interrelation of non-linear regression.

REFERENCES

[1] Приходько, С.Б., Приходько, Н.В., Фаріонова, Т.А., Ворона, М.В. (2020). Трьохфакторна нелінійна регресійна модель для оцінювання розміру Php-застосунків з відкритим кодом. *Вчені записки Таврійського національного університету імені В. І. Вернадського. Серія: Технічні науки*. 31 (70) № 1, 124-131. DOI: 10.32838/2663-5941/2020.1-1/23.

[2] Приходько, С.Б., Приходько, Н.В., Ворона, М.В., Бєловол, І.О. (2021). Нелінійна регресійна модель для оцінювання розміру web-застосунків, що створюються з використанням фреймворку Laravel. *Інформаційні технології та комп'ютерна інженерія*, 2021.1.

[3] Lépin , J.-F. (n.d.). PhpMetrics: Static analyzys for php. of Php. (2021). Retrieved from https://www.phpmetrics.org/documentation/index.html.

[4] Визуализация качества кода с PhpMetrics. (2021). Retrieved from https://habr.com/ru/post/254941/.

[5] PHP parser alternatives - PHP code analysis. LibHunt. (2021). Retrieved from https://php.libhunt.com/php-parser-alternatives.

Пономаренко Т., Геленко Ю., Котула В.

ПРОГРАМНИЙ ЗАСТОСУНОК ДЛЯ ПОБУДОВИ РЕГРЕСІЙНОЇ МОДЕЛІ ДЛЯ ОЦІНЮВАННЯ ТРУДОМІСТКОСТІ ВЕБ-ДОДАТКІВ PHP

Анотація. *У статті представлено опис програмного забезпечення для побудови багатофакторної регресійної моделі для оцінювання трудомісткості розробки веб застосунків на PHP.*

Ключові слова: *багатофакторна регресійна модель; веб застосунки на PHP; трудомісткість розробки.*