

## МАТЕМАТИЧНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ СКЛАДНОСТІ ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОЕКТУВАННЯ ЧЕРЕЗ СЕМАНТИКУ КЛАСІВ ВЕБЗАСТОСУНКІВ, ЩО СТВОРЮЮТЬСЯ ЗА ДОПОМОГОЮ PHP ФРЕЙМВОРКІВ

Приходько А.С.<sup>1</sup>

<sup>1</sup> Одеський національний університет імені І. І. Мечникова

<sup>1</sup> Україна, Одеса

<sup>1</sup>whiterandrek@gmail.com

**Анотація.** В роботі для оцінювання складності об'єктно-орієнтованого проектування через семантику класів вебзастосунків, що створюються за допомогою PHP фреймворків, на основі тривимірного перетворення Бокса-Кокса було побудовано математичну модель у вигляді еліпсоїду прогнозування для нормалізованих метрик WMC, RFC та LCOM.

**Ключові слова:** математична модель; еліпсоїд прогнозування; тривимірне перетворення Бокса-Кокса; вебзастосунок; PHP фреймворк.

**Вступна частина.** Як відомо [1], метрики WMC (Weighted Methods per Class), RFC (Response For a Class) та LCOM (Lack of COhesion in Methods) відносяться до другого кроку (семантики класів) об'єктно-орієнтованого проектування (ООП), оскільки WMC та RFC визначають, як об'єкти класу можуть «поводитися», коли вони отримують повідомлення, а метрика LCOM стосується упаковки даних і методів у визначенні класу, що забезпечує міру зв'язності класу. Але на практиці, як правило, ці метрики аналізуються окремо без врахування кореляції між ними, а прийнятні їх значення визначаються як на рівні класу, так і на рівні застосунку [2]. Так для класу, значення RFC зазвичай не повинно перевищувати 50, хоча допускається мати RFC до 100. Для класу рекомендується стандартне порогове значення від 0 до 50. Якщо RFC для класу великий, це означає, що існує висока складність. Наприклад, якщо виклик методу в класі може призвести до великої кількості різних викликів методів у цільовому та інших класах. Може бути важко перевірити поведінку класу та налагодити проблеми, оскільки розуміння поведінки класу вимагає глибокого розуміння потенційних взаємодій, які об'єкти класу можуть мати з рештою програмної системи. Перевага наявності набору показників полягає в тому, що існує потенціал для кількох показників однієї основної конструкції.

Зараз багато вебзастосунків створюються за допомогою різних фреймворків, які прискорюють розробку програмного забезпечення, у тому числі і PHP фреймворків, таких як CakePHP, Laravel, Symfony, CodeIgniter та Yii. А це потребує побудови відповідних математичних моделей для оцінювання складності ООП у тому числі через семантику класів вебзастосунків, що створюються за допомогою PHP фреймворків.

**Метою дослідження** є побудова математичної моделі для оцінювання складності ООП через семантику класів вебзастосунків, що створюються за допомогою PHP фреймворків.

**Основна частина.** На теперішній час відома математична модель для оцінювання складності ООП через ідентифікацію класів, яка побудована у вигляді еліпсоїду прогнозування в залежності від нормалізованих метрик WMC, DIT та NOC [3]. Беручи за основу запропонований в [2] підхід щодо оцінювання складності ООП та враховуючи [3], у якості математичної моделі для оцінювання складності ООП через семантику класів

вебзастосунків, що створюються за допомогою PHP фреймворків, використовуються еліпсоїди прогнозування в залежності від нормалізованих метрик WMC, RFC та LCOM.

Для побудови зазначеної математичної моделі за допомогою інструменту PhpMetrics (<https://phpmetrics.org/>) ми отримали дані з метрик WMC, RFC та LCOM для 125 вебзастосунків, що були створені за допомогою наступних популярних PHP фреймворків: CakePHP, Laravel, Symfony, CodeIgniter та Yii. Ці застосунки розміщені на сайті GitHub. Далі ми перевірили тривимірні дані на нормальність за допомогою тесту Мардіа, який базується на багатовимірних асиметрії та ексцесі. Тест показав, що розподіл даних не є гаусівським. Тому для визначення наявності викидів у даних ми використовували тест на основі квадрату відстані Махаланобіса для нормалізованих даних [4]. Як і в [3, 5], нормалізацію даних ми здійснювали за допомогою тривимірного перетворення Бокса-Кокса, оцінювання параметрів якого виконувалося за методом максимальної правдоподібності. В результаті дві точки даних виявилася тривимірними викидами – це API модуля форуму Yii2 (<https://github.com/yii-podium/yii2-api>) та застосунок Grocery CRUD (<https://github.com/scoumbourdis/grocery-crud>), який є бібліотекою PHP і фреймворку Codeigniter, яка створює повнофункціональну систему CRUD без необхідності налаштування JavaScript або CSS. Ці дві точки даних були відкинуті і остаточно ми побудували математичну модель для відповідних метрик з 123 вебзастосунків. Для вебзастосунків, що створюються за допомогою PHP фреймворків, невисока складність ООП через семантику класів характеризується виконанням наступної умови: квадрат відстані Махаланобісу для нормалізованих за допомогою тривимірного перетворення Бокса-Кокса метрик WMC, RFC та LCOM на рівні застосунку менше або дорівнює 7,82. У разі, якщо зазначений квадрат відстані Махаланобісу більше 7,82 та менше або дорівнює 12,84, то це висока складність ООП через семантику класів. Якщо вказаний квадрат відстані Махаланобісу більше за 12,84 (значення квантилю розподілу Хі-квадрат для трьох ступенів свободи та рівня значимості 0,005), то у цьому разі отримана математична модель не може бути застосована для відповідного оцінювання. Зазначимо, що параметри тривимірного перетворення Бокса-Кокса для нормалізації метрик WMC, RFC та LCOM відповідно дорівнюють -0,28340, 0,052513 та -0,75631. Для обчислення значення квадрату відстані Махаланобіса для нормалізованих за допомогою тривимірного перетворення Бокса-Кокса метрик WMC, RFC та LCOM їх вибірккові середні відповідно дорівнюють 1,629, 2,431, 0,547, а зворотня матриця до коваріаційної матриці є такою:

$$\mathbf{S}_N^{-1} = \begin{pmatrix} 20.7727 & -17.6004 & -7.29516 \\ -17.6004 & 21.7157 & -1.29931 \\ -7.29516 & -1.29931 & 50.9796 \end{pmatrix}.$$

Застосовуючи побудовану математичну модель для відповідних метрик з 123 вебзастосунку, маємо наступні результати щодо оцінювання складності ООП через семантику класів: 117 застосунків мають невисоку складність ООП, а складність 6 застосунків (Infinitas, manga-tracker, mapos, apiato, laraadmin та microweber) є високою. Infinitas (<https://github.com/infinitas/infinitas>) – це фреймворк керування вмістом із відкритим вихідним кодом, який розроблений з використанням CakePHP (v2.x). Серед іншого, це блог, cms, система розсилки інформаційних бюлетенів та онлайн-кошик для покупок. Manga-tracker – це міжсайтовий манга-трекер, а mapos (<https://github.com/RamonSilva20/mapos>) – це система контролю виконання робіт, які розроблені за допомогою фреймворку CodeIgniter.

Apiato (<https://github.com/apiato/apiato>) – це платформа для створення масштабованих і тестованих програм, орієнтованих на API, за допомогою PHP, створених на основі Laravel. LaraAdmin (<https://github.com/dwijitsolutions/laraadmin>) – це панель/CMS адміністратора Laravel з відкритим кодом, яку можна використовувати як бекенд адміністратора, інструмент керування даними або шаблон CRM для Laravel із такими функціями, як Advanced CRUD Generation, Module Manager, Backups та багато іншими. Microweber (<https://github.com/microweber/microweber>) – це конструктор веб-сайтів з функцією перетягування та CMS з електронною комерцією, який створений на основі Laravel.

**Висновки.** В роботі на основі тривимірного перетворення Бокса-Кокса було побудовано математичну модель для оцінювання складності ООП через семантику класів вебзастосунків, що створюються за допомогою PHP фреймворків. Ця модель представляє собою два еліпсоїди прогнозування в залежності від нормалізованих метрик WMC, RFC та LCOM. Ліві частини рівнянь цих еліпсоїдів однакові і містять квадрат відстані Махаланобісу для нормалізованих метрик WMC, RFC та LCOM. Ці рівняння розрізняються правими частинами, які відповідно дорівнюють 7,82 та 12,84. В подальшому планується побудова математичних моделей для оцінювання складності ООП через зв'язки між класами вебзастосунків, що створюються за допомогою PHP фреймворків.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Chidamber, S., & Kemerer, C. (1994). A metrics suite for object oriented design. *IEEE Transactions on Software Engineering*, 20(6), 476-493.
- [2] Prykhodko, S., Prykhodko, N., & Smykodub, T. (2022). A Joint Statistical Estimation of the RFC and CBO Metrics for Open-Source Applications Developed in Java. *Computer Sciences and Information Technologies : 2022 IEEE 17th International Conference (CSIT)* (pp. 442-445). Lviv, Ukraine. DOI: <https://doi.org/10.1109/CSIT56902.2022.10000457>
- [3] Prykhodko A.S., & Malakhov E.V. (2024). Determining object-oriented design complexity due to the identification of classes of open-source web applications created using PHP frameworks. *Radio Electronics, Computer Science, Control*. 69(2), 160-166. DOI: <https://doi.org/10.15588/1607-3274-2024-2-16>
- [4] Prykhodko, S., Prykhodko, N., Makarova, L., & Pukhalevych, A. (2018). Application of the squared Mahalanobis distance for detecting outliers in multivariate non-Gaussian data. *Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering : 2018 14th International Conference (TCSET)* (pp. 962-965). Lviv-Slavske, Ukraine. DOI: <https://doi.org/10.1109/TCSET.2018.8336353>
- [5] Prykhodko, S., & Prykhodko, N. (2022). A Technique for Detecting Software Quality Based on the Confidence and Prediction Intervals of Nonlinear Regression for RFC Metric. *Computer Sciences and Information Technologies : 2022 IEEE 17th International Conference (CSIT)* (pp. 499-502). Lviv, Ukraine. DOI: <https://doi.org/10.1109/CSIT56902.2022.10000532>

Prykhodko A.S.

**A mathematical model for assessing the complexity of object-oriented design through the semantics of classes of web applications created using PHP frameworks**

**Abstract.** In the work, to assess the complexity of object-oriented design through the semantics of classes of web applications created using PHP frameworks, a mathematical model in the form of a prediction ellipsoid for normalized WMC, RFC, and LCOM metrics was built based on the three-variate Box-Cox transformation.

**Keywords:** mathematical model; prediction ellipsoid; three-variate Box-Cox transformation; web application; PHP framework.