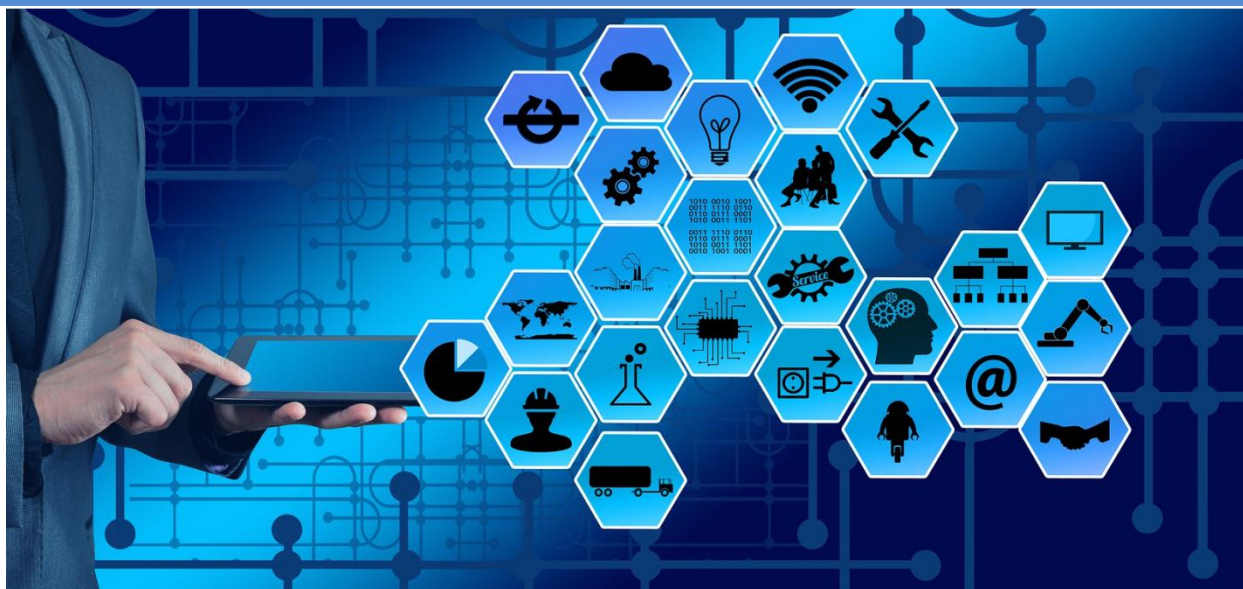




ITMAS – 2022

26-28
ЖОВТНЯ
2022

ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ: МОДЕЛІ, АЛГОРИТМИ, СИСТЕМИ



ЗБІРНИК МАТЕРІАЛІВ КОНФЕРЕНЦІЇ

Національний університет
кораблебудування імені
адмірала Макарова

Миколаїв 2022

ITMAS – 2022
Mykolaiv, Ukraine, Oct. 26-28, 2022

Міністерство освіти і науки України
Інститут модернізації змісту освіти МОН України
Національний університет кораблебудування імені адмірала Макарова

III Всеукраїнська науково-практична інтернет конференція
ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ:
МОДЕЛІ, АЛГОРИТМИ, СИСТЕМИ (ITMAS – 2022)

26-28 жовтня
Миколаїв 2022

Інформаційні технології: моделі, алгоритми, системи (ITMAS – 2022):
Матеріали III Всеукраїнської науково-практичної інтернет конференції
(26-28 жовтня 2022 р.). – Миколаїв: НУК імені адмірала Макарова, 2022. – 79 с.
Режим доступу: <http://itconf.nuos.edu.ua/2022/proceedings/>

ЗМІСТ

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ В ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЯХ....	5
Приходько С.Б., Резниченко М.В. МАТЕМАТИЧНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ СКЛАДНОСТІ ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОЕКТУВАННЯ ЧЕРЕЗ ЗВ'ЯЗКИ МІЖ КЛАСАМИ ЗАСТОСУНКІВ З ВІДКРИТИМ КОДОМ НА JAVA.....	6
Приходько С.Б., Позняков Р.О. НЕЛІНІЙНА РЕГРЕСІЙНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ МЕТРИКИ RFC ЗАСТОСУНКІВ З ВІДКРИТИМ КОДОМ НА JAVA.....	8
Латанська Л.О., Кольцов А.В. НЕЛІНІЙНА РЕГРЕСІЙНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ КІЛЬКОСТІ СТРОК КОДУ WEB-ЗАСТОСУНКІВ, ЩО СТВОРЮЮТЬСЯ З ВИКОРИСТАННЯМ PHP ФРЕЙМВОРКУ SYMFONY.....	10
Гусева-Божаткіна В.А. НЕЛІНІЙНЕ РЕГРЕСІЙНЕ РІВНЯННЯ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ COBOL – ЗАСТОСУНКІВ ДЛЯ ПЛАТФОРМИ MAINFRAME.....	12
Приходько С.Б., Пухалевич А.В., Машкін О.О. МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ТРУДОМІСТКОСТІ РОЗРОБКИ ПРОГРАМНИХ ПРОЄКТІВ ЗА НЕЛІНІЙНОЮ РЕГРЕСІЙНОЮ МОДЕЛЛЮ	16
Приходько С.Б., Пухалевич А.В., Сорокіна Є.В. МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ТРУДОМІСТКОСТІ AGILE ТЕСТУВАННЯ НЕВЕЛИКИХ WEB ПРОЄКТІВ ЗА НЕЛІНІЙНОЮ РЕГРЕСІЙНОЮ МОДЕЛЛЮ.....	18
Фаріонов М.В. ВИКОРИСТАННЯ МЕТОДІВ РОЗКРОЮ-ПАКУВАННЯ ПРИ ВИРОБНИЦТІ ВІТРИЛ	20
Давлатова Д.Х., Макарова Л.М. РЕГРЕСІЙНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ РОЗМІРУ WEB-ЗАСТОСУНКІВ, ЩО СТВОРЮЮТЬСЯ МОВОЮ PYTHON З ВИКОРИСТАННЯМ DJANGO REST FRAMEWORK.....	24
Телехан А.М., Макарова Л.М. НЕЛІНІЙНА РЕГРЕСІЙНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ РОЗМІРУ ПЕРСОНАЛЬНИХ ЗАСТОСУНКІВ-ОРГАНАЙЗЕРІВ, ЩО СТВОРЮЮТЬСЯ МОВОЮ KOTLIN ДЛЯ АНДРОЇД.....	27
Лук'яненко П.В. МАТЕМАТИЧНІ МОДЕЛІ ТА ОСОБЛИВОСТІ ЧИСЕЛЬНИХ РОЗРАХУНКІВ ПРИ РОЗРОБЦІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДОСЛІДЖЕННЯ ДИНАМІЧНИХ ВЛАСТИВОСТЕЙ ЕЛЕКТРОПРИВОДІВ	29
Дорогань І.Ю., Пухалевич А.В. МАТЕМАТИЧНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ РОЗМІРУ WORDPRESS ТЕМ НА PHP	33
Трухов А.С., Приходько С.Б. ЗАСТОСУВАННЯ КВАДРАТА ВІДСТАНІ МАХАЛАНОВІСА В ЗАДАЧІ РОЗПІЗНАВАННЯ ОБЛИЧ	35
Стрельніков М.Д., Партаєв В.К. АЛГОРИТМ КРИПТОГРАФІЧНОГО ПРОТОКОЛУ АВТЕНТИФІКАЦІЇ БЕЗ ПЕРЕСИЛАННЯ ПАРОЛЮ МЕРЕЖЕЮ У ЯВНОМУ ВИГЛЯДІ	38

Приходько С.Б., Шемчук С.Т., Карлюченко Є.К. НЕЛІНІЙНА РЕГРЕСІЙНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ КІЛЬКОСТІ СТРОК КОДУ ВЕБ ЗАСТОСУНКІВ, ЩО СТВОРЮЮТЬСЯ ЗА ДОПОМОГОЮ ФРЕЙМВОРКУ DJUNGO	44
Латанська Л.О., Бринзей О.А. НЕЛІНІЙНА РЕГРЕСІЙНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ РОЗМІРУ СЕРВЕРНИХ ЗАСТОСУНКІВ, ЩО СТВОРЮЮТЬСЯ З ВИКОРИСТАННЯМ ФРЕЙМВОРКУ NESTJS	46
Латанська Л.О., Борецько А.В. ТРЬОХФАКТОРНА НЕЛІНІЙНА РЕГРЕСІЙНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ РОЗМІРУ WEB-ЗАСТОСУНКІВ НА JAVA З ВИКОРИСТАННЯМ SPRING FRAMEWORK	48
Приходько С.Б., Папакін М.М. МАТЕМАТИЧНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ СКЛАДНОСТІ ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОЕКТУВАННЯ ЧЕРЕЗ ЗВ'ЯЗКИ МІЖ КЛАСАМИ КОМП'ЮТЕРНИХ ІГОР	50
Приходько А.С. АНАЛІТИЧНА МОДЕЛЬ ЧОТИРИВИМІРНОГО ЗАКОНУ РОЗПОДІЛУ МЕТРИК ВЕБ ЗАСТОСУНКІВ, ЩО СТВОРЮЮТЬСЯ ЗА ДОПОМОГОЮ ФРЕЙМВОРКУ САКЕРНР	52
УПРАВЛІННЯ ПРОЕКТАМИ І ПРОГРАМАМИ	54
Вербицький В.О., Гусєва Ю.Ю. ПРОГРАМНІ ЗАСОБИ МОНІТОРИНГУ ПРОГРЕСУ ПРОЄКТУ В УМОВАХ НЕПОВНОЇ ІНФОРМАЦІЇ	55
Гусєва-Божаткіна В.А., Єжгуров Р.В. НАПРЯМКИ РОЗВИТКУ УПРАВЛІННЯ ПРОЄКТАМИ МАРКЕТИНГ-ЛОГІСТИЧНОГО ЗАБЕЗПЕЧЕННЯ ЛАНЦЮГІВ ПОСТАЧАННЯ ФАРМАЦЕВТИЧНИХ ТОВАРІВ	58
ІНФОРМАЦІЙНІ СИСТЕМИ ТА ТЕХНОЛОГІЇ	61
Маршак О.І., Бобровський О.С., Коваль С.С., Ворона М.В. ДОСЛІДЖЕННЯ ПРОЦЕСІВ ІНФОРМАЦІЙНОЇ ПІДТРИМКИ ОСВІТНЬОЇ ДІЯЛЬНОСТІ	62
Фаріонова Т.А., Орехов О.С. ПРОЄКТ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ОЦІНКИ ВПЛИВУ УЛЬТРАФІОЛЕТОВОГО ВИПРОМІНЮВАННЯ НА ЕКІПАЖ СУДНА	65
Мещеряков В.І., Устенко А.С. АНАЛІЗ ІСНУЮЧИХ МАТЕМАТИЧНИХ МОДЕЛЕЙ ПРОГНОЗУВАННЯ ПРОДУКТИВНОСТІ ТА НАДІЙНОСТІ АКУМУЛЯТОРНИХ БАТАРЕЙ ЕЛЕКТРОМОБІЛЕЙ	68
Новожилова О.В. КОНЦЕПЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ОБРОБКИ СТАТИСТИЧНИХ ДАНИХ НАУКОВИХ ДОСЛІДЖЕНЬ	71
Гайдаєнко О.В., Бандурко К.Ю., Світенко І.В. ДОСЛІДЖЕННЯ МЕТОДІВ ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ НАВЧАННЯ	74
Макарова Л.М., Камінський С.С., Бризгалов М.В. ВИКОРИСТАННЯ ЕНТРОПІЇ ДЛЯ ЗНАХОДЖЕННЯ ВНЕСЕННЯ ЗМІН КОДУ ВИКОНУВАНИХ ФАЙЛІВ	77



МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ В ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЯХ

УДК 004.412:519.237.5

МАТЕМАТИЧНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ СКЛАДНОСТІ ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОЕКТУВАННЯ ЧЕРЕЗ ЗВ'ЯЗКИ МІЖ КЛАСАМИ ЗАСТОСУНКІВ З ВІДКРИТИМ КОДОМ НА JAVA

Приходько С.Б., д.т.н., професор¹; Резниченко М.В.²

^{1,2} Національний університет кораблебудування імені адмірала Макарова

^{1,2} Україна, Миколаїв

¹sergiy.prykhodko@nuos.edu.ua; ²211265@nuos.edu.ua

Анотація. В роботі на основі перетворення у вигляді десяткового логарифму побудовано еліпс прогнозування для оцінювання складності об'єктно-орієнтованого проектування через зв'язки між класами застосунків з відкритим кодом на Java.

Ключові слова: еліпс прогнозування; оцінювання складності об'єктно-орієнтованого проектування; зв'язки між класами; застосунок з відкритим кодом; Java.

Вступна частина. На сьогодні задача оцінювання складності об'єктно-орієнтованого проектування (ООП) через зв'язки між класами є важливою для ІТ-компаній, оскільки тісно пов'язана з подальшими витратами, які виникають в процесі життєвого циклу програмного забезпечення. Згідно з [1] для оцінювання складності ООП через зв'язки між класами можуть бути використані дві програмні метрики: зв'язаність між об'єктами (CBO, Coupling Between Objects) та відгук класу (RFC, the Response for Class). Але на практиці, як правило, ці метрики аналізуються окремо без врахування кореляції між ними, а прийнятні їх значення визначаються лише на рівні класу. Так для класу, значення CBO між 1 та 5 вважається добрим, а рекомендоване значення RFC складає від 0 до 50. Відповідно до [2], значення CBO більше 14 є занадто високим. Однак існують застосунки на Java, у яких значення CBO для деяких класів значно перевищують 4 і навіть 14. Наприклад, у застосунку TuxGuitar версії 1.5.2, програмні метрики якого проаналізовано в [3], є клас `org.herac.tuxguitar.app.action.installer.TGActionInstaller`, де значення CBO дорівнює 252. Зауважте, що TuxGuitar – це досить велика програма з 207810 рядками коду, що включає 2381 клас. Загальний CBO для всіх класів становить 20189. Тому виникає потреба у математичній моделі, яка би дозволяла здійснювати оцінювання складності ООП через зв'язки між класами за значеннями CBO та RFC з урахуванням їх кореляції на рівні застосунку в цілому, включаючи і застосунки з відкритим кодом на Java.

Метою дослідження є побудова математичної моделі для оцінювання складності ООП через зв'язки між класами в залежності від метрик RFC та CBO.

Основна частина. Як і в [4], у якості такої математичної моделі для оцінювання складності ООП через зв'язки між класами пропонується використовувати рівняння трансформованого еліпсу передбачення в залежності від метрик RFC та CBO. Як відомо [4], еліпс передбачення використовують для визначення викидів у двовимірних гаусівських даних. Але двовимірний розподіл даних метрик RFC та CBO для Java-застосунків не є гаусівським. Так, ми перевірили двовимірні дані метрик RFC та CBO з 44 Java-застосунків з відкритим кодом, які розміщені на сайті GitHub (<https://github.com>), на нормальність за допомогою критерію Мардіа (Mardia's test), якій базується на багатовимірних асиметрії та ексцесі. За цим критерієм двовимірний розподіл зазначених даних не є гаусівським. Тому, як і в [4], ми використовуємо трансформований еліпс передбачення, який будується наступним

чином. Спочатку за допомогою взаємо-зворотного нормалізуючого перетворення двовимірні негаусівські дані нормалізують таким чином щоб розподіл нормалізованих даних був наближений до гаусівського. Далі для нормалізованих даних будують еліпс передбачення. І нарешті, за еліпсом передбачення для нормалізованих даних із використанням зворотного до нормалізуючого перетворення отримують трансформований еліпс передбачення. Вихід точки негаусівських даних за трансформований еліпс передбачення буде вказувати на наявність викиду з певною довірчою ймовірністю, для якої цей еліпс було побудовано. Також вихід точки даних із значеннями метрик RFC та СВО на рівні застосунку за еліпс буде вказувати і на високу складність через зв'язки між класами для цього застосунку. І навпаки, якщо зазначена точка знаходиться всередині за еліпсу, то вказана складність є невисокою. При цьому значення довірчої ймовірності ми пропонуємо брати 0,995, яке зазвичай обирають при визначенні викидів у багатовимірних даних. Для нормалізації двовимірних даних метрик RFC та СВО з 44 Java-застосунків з відкритим кодом ми використовували перетворення у формі десятичного логарифму. На основі побудованого рівняння трансформованого еліпсу передбачення виявлено, що три популярні застосунки з відкритим кодом на Java (FreeMind, jEdit та TuxGuitar), метрики яких аналізуються в [3], мають невисоку складність. При цьому у зазначеному рівнянні еліпсу квадрат відстані Махаланобісу (Mahalanobis) для десятичних логарифмів метрик RFC та СВО на рівні застосунку дорівнює 12,62.

Висновки. У якості математичної моделі для оцінювання складності ООП через зв'язки між класами запропоновано використовувати рівняння трансформованого еліпсу передбачення, яке було побудоване на основі перетворення у формі десятичного логарифму. Для Java-застосунків невисока складність ООП через зв'язки між класами характеризується виконанням наступної умови: квадрат відстані Махаланобісу для десятичних логарифмів метрик RFC та СВО на рівні застосунку менше або дорівнює 12,62.

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] Sahraoui, H.A., Godin, R., & Miceli, T. (2000). Can metrics help to bridge the gap between the improvement of OO design quality and its automation? *Software Maintenance : 2000 International Conference (ICSM)* (pp. 154-162). San Jose, CA, USA. DOI: <https://doi.org/10.1109/ICSM.2000.883034>
- [3] Molnar, A.J., Neamțu, A., & Motogna, S. (2020). Evaluation of software product quality metrics. In E. Damiani, G. Spanoudakis, & L. Maciaszek (eds). *Evaluation of Novel Approaches to Software Engineering : 14th International Conference (ENASE 2019), Communications in Computer and Information Science: Vol. 1172* (pp. 163-187). Springer, Cham. DOI: https://doi.org/10.1007/978-3-030-40223-5_8
- [4] Приходько, С.Б., Смикодуб, Т.Г. (2021). Математична модель для перевірки взаємозв'язків між частинами Java-застосунків за метриками RFC та СВО. *Інформаційні технології: моделі, алгоритми, системи : II-га Всеукраїнська науково-практична інтернет конференція (ITMAS – 2021)* (С. 25-26). Миколаїв, Україна. <http://itconf.nuos.edu.ua/2021/proceedings/>

Prykhodko S.B, Reznichenko M.V.

A mathematical model for evaluating the complexity of object-oriented design due to the relationships between classes of open-source applications in Java

Abstract. In the work, based on the transformation in the form of a decimal logarithm, a prediction ellipse is built for evaluating the complexity of object-oriented design due to the relationships between classes of open-source applications in Java.

Keywords: complexity; object-oriented design; open-source app; Java; software metric; prediction ellipse.

УДК 004.412:519.237.5

НЕЛІНІЙНА РЕГРЕСІЙНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ МЕТРИКИ RFC ЗАСТОСУНКІВ З ВІДКРИТИМ КОДОМ НА JAVA

Приходько С.Б., д.т.н., професор¹; Позняков Р.О.²

^{1,2} Національний університет кораблебудування імені адмірала Макарова

^{1,2} Україна, Миколаїв

¹sergiy.prykhodko@nuos.edu.ua; ²211264@nuos.edu.ua

Анотація. В роботі на основі перетворення у вигляді десяткового логарифму побудована двохфакторна нелінійна регресійна модель для оцінювання метрики RFC застосунків з відкритим кодом на Java в залежності від метрик CBO та WMC.

Ключові слова: оцінювання, метрика RFC; застосунок з відкритим кодом; Java; нелінійна регресійна модель.

Вступна частина. Для визначення якості програмного забезпечення (ПЗ) запропоновано використовувати різні метрики ПЗ, включаючи RFC (response for a class), CBO (coupling between object classes) та WMC (weighted methods per class) на рівні програми [1]. Однак відомі лише точкові оцінки їх середніх для виявлення якості програмних систем [1]. А інтервальних оцінок для показників ПЗ на рівні програми для визначення її якості немає. Також відомо [1], що метрика RFC залежить від метрик CBO та WMC. У [1] автори побудували рівняння лінійної регресії для оцінювання середнього RFC. Водночас не завжди є теоретичне обґрунтування використання моделей лінійної регресії, наприклад, коли розподіл залишків не є гаусовим [2]. У цьому випадку необхідно застосовувати нелінійні регресійні моделі [3].

Метою дослідження є побудова нелінійної регресійної моделі для оцінювання метрики RFC застосунків з відкритим кодом на Java в залежності від метрик CBO та WMC.

Основна частина. Розглянемо побудову нелінійної регресійної моделі для оцінювання метрики RFC застосунків з відкритим кодом на Java в залежності від метрик CBO та WMC. Для цього зазначені три метрики з 49 застосунків з відкритим кодом на Java, які розміщені на сайті GitHub (<https://github.com>), були отримані за допомогою інструменту СК (<https://github.com/mauricioaniche/ck>). Далі ми перевірили тривимірні дані на нормальність за допомогою тесту Мардіа (Mardia's test), 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 за десятковим логарифмом даних.

Використовуючи зібрані дані із зазначених вище метрик застосунків з відкритим кодом на Java спочатку була побудована лінійна регресійна модель $Y = \hat{b}_0 + \hat{b}_1 X_1 + \hat{b}_2 X_2 + \varepsilon$, де $\hat{b}_0$, $\hat{b}_1$ та $\hat{b}_2$ є оцінками параметрів. Змінна ε , яка описує залишки, повинна бути випадковою величиною з розподілом Гаусу. Нульова гіпотеза про нормальність розподілу ε була перевірена за критерієм Пірсона. Ця гіпотеза була відкинута для рівня значимості 0,05. Те, що розподіл ε не є гаусівським вказує на відсутність теоретичного обґрунтування застосування лінійної регресійної моделі та на необхідність побудови нелінійної регресійної моделі. Нелінійна регресійна модель, яка була побудована на основі десяткового логарифму має вигляд $Y = 10^{\varepsilon + \hat{b}_0} X_1^{\hat{b}_1} X_2^{\hat{b}_2}$, де ε – гаусівська випадкова величина з нульовим математичним сподіванням та середньоквадратичним відхиленням, оцінка якого дорівнює

0,06538, $\hat{b}_0$, $\hat{b}_1$ та $\hat{b}_2$ є оцінками параметрів, які мають відповідно такі значення: -0,054776, 0,672260 та 0,441541. Оцінки параметрів були отримані за допомогою методу найменших квадратів. Ця модель була побудована за методом [3] із врахуванням наявності викидів за даними з 46 застосунків, що залишилися, і може застосовуватися для значень метрик, для яких квадрат відстані Махаланобісу їх десяткових логарифмів менше або дорівнює 12,84.

Створена нелінійна регресійна модель була перевірена за множинним коефіцієнтом детермінації R^2 , середньою величиною відносної помилки MMRE і відсотком прогнозованих результатів, для яких величини відносної помилки MRE менші за 0,25, PRED(0,25). Допустимі значення MMRE і PRED(0,25) складають не більше 0,25 і не менше 0,75 відповідно. Значення R^2 , MMRE і PRED(0,25) для створеної моделі дорівнюють відповідно 0,8718, 0,1262 і 0,9783. Це свідчить про добру якість отриманої моделі. За побудованою нелінійною регресійною моделлю виконано оцінювання метрики RFC для трьох популярних застосунків з відкритим кодом на Java (FreeMind, jEdit та TuxGuitar), метрики яких аналізуються в [4] на рівні застосунку. Для FreeMind, jEdit та TuxGuitar значення оцінок метрики RFC дорівнюють 8,316, 7,813 та 10,199 відповідно. А актуальні значення RFC за [4] є такими 13,54, 10,62 та 17,78 відповідно. Отриманий результат вказує на те, що лише для jEdit відносна похибка (MRE) менше за 30%, а саме – 26,4%. Для застосунків FreeMind та TuxGuitar значення MRE складає 38,6% та 42,6% відповідно. Такий результат можна пояснити тим, що значення квадрату відстані Махаланобісу для десяткових логарифмів метрик застосунків FreeMind, jEdit та TuxGuitar більші за 12,84, а саме дорівнюють 17,11, 13,59 та 16,16 відповідно. Це вказує і на необхідність у подальшому використання більш складних перетворень, наприклад, тривимірних перетворень Джонсона або Бокса-Кокса.

Висновки. В роботі на основі десяткового логарифму була побудована нелінійна регресійна модель для оцінювання метрики RFC застосунків з відкритим кодом на Java в залежності від метрик CBO та WMC, яка має задовільну якість відповідного оцінювання.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Laing, V., & Coleman, C. (2001). *Principal Components of Orthogonal Object-Oriented Metrics*. White Paper Analyzing Results of NASA Object-Oriented Data (Task 323-08-14). Greenbelt, Maryland (US): NASA Goddard Space Flight Center, the Software Assurance Technology Center (SATC).
- [2] Bates, D.M., & Watts, D.G. (1988). *Nonlinear Regression Analysis and Its Applications* (2nd ed.). New York: John Wiley & Sons. <http://dx.doi.org/10.1002/9780470316757>
- [3] Prykhodko, S., & Prykhodko, N. (2022). A modified technique for constructing nonlinear regression models based on the multivariate normalizing transformations. *Selected Papers of the VIII International Scientific Conference "Information Technology and Implementation" (IT&I-2021). Workshop Proceedings (IT&I-WS-2021)*, Kyiv, Ukraine, December 1-3, 2021. *CEUR Workshop Proceedings*, 3179, 156-166. http://ceur-ws.org/Vol-3179/Paper_15.pdf
- [4] Molnar, A.J., Neamtu, A., & Motogna, S. (2020). Evaluation of software product quality metrics. In E. Damiani, G. Spanoudakis, & L. Maciaszek (eds). *Evaluation of Novel Approaches to Software Engineering : 14th International Conference (ENASE 2019), Communications in Computer and Information Science: Vol. 1172* (pp. 163-187). Springer, Cham. DOI: https://doi.org/10.1007/978-3-030-40223-5_8

Prykhodko S.B., Poznyakov R.O.

A nonlinear regression model for estimating the RFC metric of open-source applications in Java

Abstract. In work, a nonlinear regression model with two predictors for estimating the RFC metric of open-source applications in Java depending on CBO and WMC metrics has been built.

Keywords: estimation; RFC metric; open-source app; nonlinear regression model.

УДК 004.412:519.237.5

НЕЛІНІЙНА РЕГРЕСІЙНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ КІЛЬКОСТІ СТРОК КОДУ WEB-ЗАСТОСУНКІВ, ЩО СТВОРЮЮТЬСЯ З ВИКОРИСТАННЯМ PHP ФРЕЙМВОРКУ SYMFONY

Латанська Л.О., к.ф.-м.н., доцент¹; Кольцов А.В.²

^{1,2} Національний університет кораблебудування імені адмірала Макарова

^{1,2} Україна, Миколаїв

¹ llatanskaya@gmail.com; ² andrew.koltsv@gmail.com

Анотація. В роботі було побудовано нелінійну регресійну модель для оцінювання кількості строк коду web-застосунків, що створюються з використанням фреймворку Symfony.

Ключові слова: оцінка кількості строк коду програмного забезпечення; web-застосунок; нелінійна регресійна модель.

Вступна частина. Проведення оцінювання розміру проектів програмного забезпечення на початкових стадіях є важливим питанням у галузі інженерії програмного забезпечення, адже отримані оцінки можна застосувати для прогнозування трудомісткості за допомогою відомих моделей як, наприклад, COSOMO II. Це викликає необхідність дослідження, побудови та удосконалення моделей для оцінювання розміру програмного забезпечення.

Значна частина проектів програмного забезпечення сьогодні припадає на web-застосунки. Мова PHP сьогодні застосовується приблизно в 71,4% сайтів і більша частина сайтів використовує саме фреймворки мови PHP, зокрема Symfony, стали популярніть якого підтверджують дослідження у 2015-2021 роках [1-3].

Серед регресійних моделей для оцінювання розміру програмних застосунків на PHP відомі як лінійна модель [4], що побудована за допомогою множинного лінійного регресійного аналізу, так і група нелінійних багатофакторних регресійних моделей [5, 6]. Моделі [4, 5] будуються на основі таких трьох метрик концептуальної моделі, як загальна кількість класів, загальна кількість зв'язків, середня кількість атрибутів на клас, тоді як у роботі [6] було запропоновано наступний перелік метрик: кількість класів; сума середньої кількості класів, на які впливає даний клас, і середньої кількості класів, з яких даний клас отримує ефекти; середня кількість методів. Для PHP фреймворків CakePHP та Yii відомі якісні моделі [7, 8], що використовують такі метрики як: кількість класів, середня кількість методів на кожен клас, глибина дерева наслідування. Вищенаведене слугує підставою для дослідження та побудови регресійних моделей для фреймворку Symfony.

Метою роботи є побудова нелінійної регресійної моделі для оцінювання розміру web-застосунків, що створюються з використанням фреймворку Symfony, у тисячах строк коду з використанням нормалізуючого перетворення десяткового логарифму.

Основна частина. Побудова нелінійної регресійної моделі на першому етапі вимагає пошуку емпіричних даних. Серед розміщених на сайті GitHub (<https://github.com/>) web-застосунків, було знайдено 50 проектів, що створюються з використанням фреймворку Symfony. Для отримання метрик програмного забезпечення було використано інструмент PhpMetrics. Для побудови нелінійної регресійної моделі використовувались метрики діаграми класів такі, як і в [7-8]: розмір web-застосунку Y у тисячах строк коду, кількість класів X_1 , середня кількість методів у класі X_2 , глибина дерева наслідування X_3 . Наведений

перелік метрик було обрано через ряд причин, серед яких: можливість отримання цих метрик на ранній стадії роботи над проектом з діаграми класів, відсутність мультиколінеарності між метриками, високі показники якості моделей, отримані в роботах [7-8]. Для визначення викидів у даних використовували квадрат відстані Махаланобіса.

Отримана нелінійна регресійна модель для web-застосунків, що створюються з використанням фреймворку Symfony, має вигляд: $Y = 10^{\varepsilon + \hat{b}_0} X_1^{\hat{b}_1} X_2^{\hat{b}_2} X_3^{\hat{b}_3}$, де ε – гаусівська випадкова величина. Оцінки параметрів $\hat{b}_0, \hat{b}_1, \hat{b}_2, \hat{b}_3$ дорівнюють -1,6378, 0,9664, 0,9630 та 0,4627 відповідно. Якість побудованої моделі було оцінено за допомогою розрахунку множинного коефіцієнту детермінації R^2 , середньої величини відносної похибки $MMRE$ та відсотку прогнозованих результатів $PRED(0,25)$. Значення R^2 , $MMRE$ і $PRED(0,25)$ для створеної моделі дорівнюють відповідно 0,900, 0,187 і 0,756, що свідчить про прийнятну якість отриманої моделі.

Висновки. В роботі було побудовано нелінійну регресійну модель, для оцінювання розміру web-застосунків, що створюються з використанням фреймворку Symfony, яка залежить від трьох факторів діаграми класів. Отримана модель має задовільні оцінки якості, що дозволяє використовувати її для оцінювання розміру web-застосунків, що створюються з використанням фреймворку Symfony.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] W3Techs Web Technology Surveys. (n. d.). *Usage statistics of PHP for websites*. Retrieved October 19, 2022, from <https://w3techs.com/technologies/details/pl-php>.
- [2] SitePoint Survey Results. (2015, March 30). *The Best PHP Framework for 2015*. <https://www.sitepoint.com/best-php-framework-2015-sitepoint-survey-results/>.
- [3] JetBrains. (2021). *PHP developer ecosystem survey*. <https://www.jetbrains.com/lp/devecosystem-2021/php/>.
- [4] Tan, H.B.K., Zhao, Y., & Zhang, H. (2006). *Estimating LOC for information systems from their conceptual data models*, Software Engineering : the 28th International Conference (ICSE '06). Shanghai, China.
- [5] Prykhodko, N.V., & Prykhodko, S.B. (2018). Constructing the non-linear regression models on the basis of multivariate normalizing transformations. *Electronic modeling*, 6 (40), 101-110. DOI: 10.15407/emodel.40.06.101
- [6] 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. *Науковий журнал «Вчені записки Таврійського національного університету імені В. І. Вернадського. Серія: Технічні науки»*, 31 (70), 124-131. DOI: 10.32838/2663-5941/2020.1- 1/23
- [7] Prykhodko, S.B., Shutko, I.S., & Prykhodko, A.S. (2021). A nonlinear regression model to estimate the size of web apps created using the cakephp framework. *Radio Electronics Computer Science Control*, 4, 129-139. DOI: <http://dx.doi.org/10.15588/1607-3274-2021-4-12>
- [8] Prykhodko S.B., Shutko I.S., & Prykhodko A.S. (2021). Early LOC Estimation of Web Apps Created Using Yii Framework by Nonlinear Regression Models. *WSEAS Transactions on Computers*, 40, 321-328. DOI: <https://doi.org/10.37394/23205.2021.20.35>

Liudmyla Latanska, Andrii Koltsov

A nonlinear regression model for estimating LOC of web applications created using Symfony framework

Abstract. In work the nonlinear regression model for estimating LOC of web application created using Symfony framework have been created.

Keywords: LOC estimation; Web app; nonlinear regression model.

УДК 004.412

НЕЛІНІЙНЕ РЕГРЕСІЙНЕ РІВНЯННЯ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ COBOL – ЗАСТОСУНКІВ ДЛЯ ПЛАТФОРМИ MAINFRAME

Гусєва-Божаткіна В. А.

Національний університет кораблебудування імені адмірала Макарова

Україна, Миколаїв

guseva.bozhatkina@nuos.edu.ua; ORCID: 0000-0002-1117-3391

Анотація. Розроблено нелінійне регресійне рівняння для оцінювання тривалості розробки Cobol-застосунків для платформи Mainframe. Проведено оцінку існуючої моделі ISBSG для оцінювання часу розробки програмних застосунків в залежності від трудомісткості їх розробки для платформи Mainframe. Отримана модель у порівнянні з моделлю ISBSG, підвищує достовірність оцінювання тривалості розробки Cobol-застосунків для платформи Mainframe.

Ключові слова: програмний застосунок; тривалість розробки програмних застосунків; платформа Mainframe; регресійний аналіз; мова Cobol.

Вступна частина. Розробка та реінжиніринг програмних застосунків мовою програмування Cobol досі є більш актуальною темою для багатьох компаній, зокрема банків та фінустанов [1]. Варто підкреслити окремо, що мова програмування Cobol наразі не є досить поширеною і відповідного роду кваліфікації програмістів стає все менше. Поступово, у зв'язку зі збільшенням завдань від замовників, дана сфера діяльності переходить до невеликих аутсорсових організацій. Можна констатувати, що успіх програмних проєктів на початкових стадіях залежить від рішень, які приймаються проджект-менеджерами [2].

Актуальність проблеми отримання ефективної системи оцінювання тривалості розробки та реінжинірингу програмних застосунків в даний час є важливим завданням, що вимагає удосконалення існуючих методів, бо:

- за статистикою близько 30% програмних проєктів мають труднощі з реалізацією (не вкладаються в бюджет, виходять з порушенням часових обмежень, або неналежної якості);
- програмні проєкти для платформи Mainframe характеризуються своїм значним розміром, великою кількістю компонентів та спеціальними вимогами до надійності;
- частка програмного забезпечення для платформи Mainframe, основною мовою розробки якого є Cobol, складає приблизно 17,8%;
- існуючі моделі COCOMO та ISBSG не враховують особливості програмного продукту, такі як мова та засоби розробки, сфера використання. Для цих моделей немає побудованих рівнянь довірчих інтервалів та рівнянь інтервалів прогнозування тривалості.

Метою роботи є підвищення достовірності оцінювання тривалості розробки програмних застосунків мовою програмування Cobol за рахунок побудови нелінійної регресійної моделі.

Виходячи з поставленої мети та проблематики теми дослідження необхідно вирішити наступні завдання:

- провести аналіз існуючих методів та моделей оцінювання тривалості програмних застосунків;

- побудувати нелінійну регресійну модель тривалості розробки застосунків на Cobol для Mainframe в залежності від їх трудомісткості, шляхом побудови рівняння нелінійної регресії;
- на основі удосконалених моделей розробити програмне забезпечення для оцінювання тривалості розробки та Cobol-застосунків для платформи Mainframe.

У якості об'єкта дослідження було обрано процес оцінювання тривалості розробки Cobol-застосунків для платформи Mainframe.

Предметом дослідження є математична модель для оцінювання тривалості розробки Cobol-застосунків для платформи Mainframe та безпосередньо програма для їх реалізації.

Наукова новизна одержаних результатів полягає в отриманні подальшого розвитку моделі ISBSG для оцінювання тривалості розробки програмних застосунків для платформи Mainframe в залежності від трудомісткості їх розробки, за рахунок створення нелінійної регресійної моделі для Cobol-застосунків, яка побудована з урахуванням викидів. Це дозволяє підвищити достовірність оцінювання тривалості розробки Cobol- застосунків для платформи Mainframe.

Основна частина. Завдання оцінювання тривалості розробки програмних застосунків на ранній стадії є важливою задачею, оскільки ця інформація використовується для прогнозування часу розробки ПЗ за допомогою таких моделей як COCOMO II [3; 4]. Наразі існує певною мірою поширена проблема – не існує єдиного вимірювання, набору метрик та показників для оцінювання тривалості розробки ПЗ. Однак існують методи для оцінювання тривалості розробки ПЗ, деякі з них базуються на кількості строк коду, інші – обчислюють тривалість з функціональних, технічних або інших аспектів [1; 5]. Але більшість з них стають непридатними до використання через відсутність даних, ресурсів або експертних навичок в цій галузі.

Станом на сьогодні існує досить велика кількість методів для оцінювання тривалості розробки ПЗ. Нормалізуючі перетворення дуже часто є надійним способом побудови нелінійних регресійних моделей та рівнянь. Однак добре відомі методи їх побудови, що засновані на одновимірних нормалізуючих перетвореннях (таких як логарифмічне, Бокса-Кокса, Джонсона), не враховують кореляції між випадковими змінними у разі нормалізації негаусових даних. Це призводить до необхідності використання нормалізуючих перетворень, які враховують цю кореляцію, для побудови нелінійних регресійних моделей та рівнянь для оцінювання тривалості ПЗ інформаційних систем, в тому числі з відкритим кодом на Cobol [6].

Враховуючи рекомендації зі [7], загальний алгоритм процесу побудови нелінійної регресійної моделі (після процесу нормалізації емпіричних даних) має ітераційний характер та проходить в декілька етапів, що представлено у вигляді діаграми на рисунку 1.

Для побудови нелінійної регресійної моделі були використані дані за 124-ма програмними продуктами мовою програмування Cobol для платформи Mainframe з репозитарію International Software Benchmarking Standards Group (ISBSG) 2021 р. [5]. Вибір впав саме на цей репозитарій, через велику кількість програмних проєктів, що в ньому присутні, для яких вже визначена якість зібраних статистичних даних і даних щодо тривалості та трудомісткості розробки.

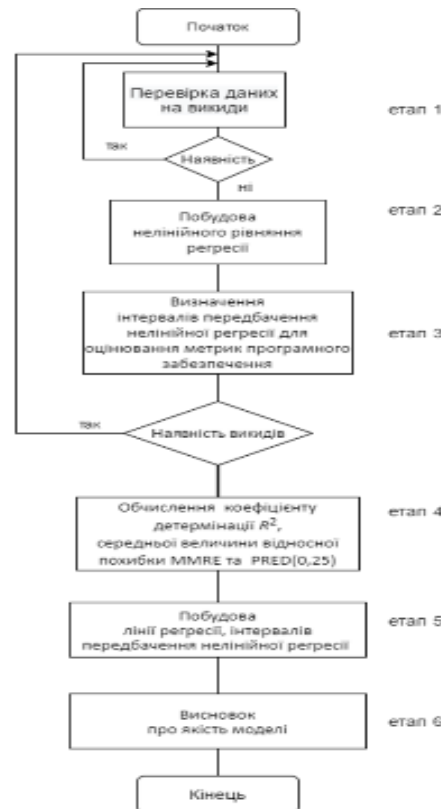


Рисунок 1 – Алгоритм побудови нелінійної регресійної моделі

На рисунку 2 зображено тривалість розробки Cobol-застосунків для платформи Mainframe в залежності від трудомісткості цих проектів за даними репозитарію ISBSG.

Проведено аналіз отриманих результатів розрахунків з моделлю ISBSG для платформи Mainframe. Показники побудованої нелінійної моделі регресії для оцінювання тривалості розробки Cobol-застосунків для платформи Mainframe наступні:

Якість побудованої моделі: $n = 124$; $R^2 = 0,3683$ $PRED(0,25) = 0,3952$; $MMRE = 0,5325$.

Якість моделі ISBSG: $n = 124$; $R^2 = 0,2962$; $PRED(0,25) = 0,4032$; $MMRE = 0,4846$.

Після відкидання викидів, зі 124 застосунків залишилось 97. Показник R^2 вищий ніж у моделі ISBSG, що говорить про те, що отримана модель більш достовірна.

Якість побудованої моделі: $n = 97$; $R^2 = 0,5743$; $PRED(0,25) = 0,4845$; $MMRE = 0,3002$.

Якість моделі ISBSG: $n = 97$; $R^2 = 0,5123$; $PRED(0,25) = 0,5155$; $MMRE = 0,2859$.

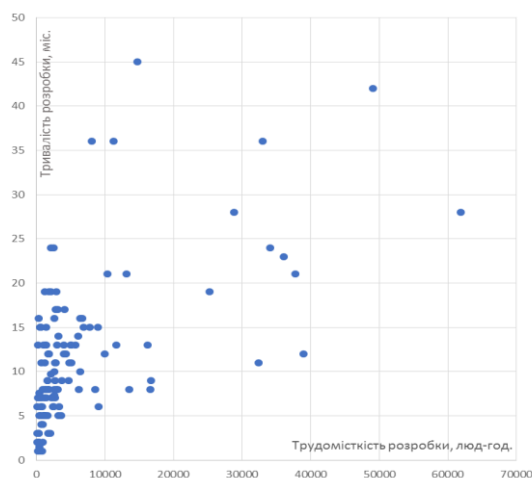


Рисунок 2 – Тривалість розробки Cobol-застосунків для платформи Mainframe в залежності від трудомісткості цих проектів за даними репозитарію ISBSG

Побудована лінійна модель регресії: $z_W = 0.312 - 0.086z_P + \varepsilon$.

Побудована нелінійна модель регресії: $P = 10^{\varepsilon - 0.086} \cdot W^{0.312}$.

Отримана модель регресії тривалості розробки застосунків, написаних мовою програмування Cobol для платформи Mainframe в залежності від трудомісткості розробки таких застосунків на основі десятичного логарифма має кращі характеристики R^2 , ніж ISBSG модель.

Проведений аналіз показав необхідність проведення подальших досліджень з метою побудови покращеної нелінійної регресійної моделі для оцінювання тривалості розробки застосунків, написаних мовою програмування Cobol для платформи Mainframe.

Висновки. Розроблена нелінійна регресійна модель тривалості розробки Cobol-застосунків для платформи Mainframe (з урахування типів ПЗ). Отримана модель у порівнянні з моделлю ISBSG, має кращі характеристики R^2 . Доцільно також розглянути можливість більш складних перетворень та дослідити вплив додаткових чинників, таких як: сфера використання програмного продукту (банківська справа, виробництво, телекомунікації тощо), функціональний розмір та ін.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Geet, J. V., Demeyer, S. (2009). Feature location in COBOL mainframe systems: an experience report. *Software Maintenance (ICSM 2009): Proc. of the 25th IEEE International Conference on*, Sept. 20–26. (2009). Edmonton, Alberta, Canada. DOI: 10.1109/ICSM.2009.5306312.
- [2] A guide to the project management body of knowledge. Fifth edition. Project Management Institute. (2013). 589 p.
- [3] Capers, J. (2013). Function points as a universal software metric. *ACM SIGSOFT Software Engineering Notes*. Vol. 38, No. 4. (P. 1–27). DOI: 10.1145/2492248.2492268.
- [4] International Software Benchmarking Standards Group (2016, March). ISBSG Data Demographics D&E Repository Release 2016 R1. <http://isbsg.org/wp-content/uploads/2016/04/ISBSG-DE-Demographics-2016-R1.pdf>.
- [5] Kissell, R. L. (2021). Nonlinear regression modeling. In book: *Algorithmic trading methods*. DOI: 10.1016/B978-0-12-815630-8.00008-9.
- [6] Sneed, H., Erdoes, K. (2013). Migrating AS400-COBOL to Java: A report from the field. *Software Maintenance and Reengineering (CSMR): Proc. of the 2013 17th European Conference*. DOI: 10.1109/CSMR.2013.32.
- [7] Prykhodko, S. B., Pukhalevich, A. V. (2014). Developing PC Software Project Duration Model based on Johnson transformation. *Proc. of the 12th Int. Conf. Modern Problems of Radio Engineering, Telecommunications and Computer Science TCSET'2014, Lviv – Slavske, Ukraine. Lviv Polytechnic National University*. P. 114–116.

Guseva-Bozhatkina V.A.

Nonlinear regression equation for estimating the development durations of Cobol – applications for the mainframe platform

Abstract. A nonlinear regression equation was developed to estimate the duration of development of Cobol applications for the Mainframe platform. An evaluation of the existing ISBSG model for estimating the development time of software applications depending on the complexity of their development for the Mainframe platform was carried out. The obtained model, in comparison with the ISBSG model, indicates the reliability of estimating the duration of development of Cobol applications for the Mainframe platform.

Keywords: Software application; duration of software application; Mainframe platform; regression analysis; Cobol language.

УДК 004.412:519.237.5

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ТРУДОМІСТКОСТІ РОЗРОБКИ ПРОГРАМНИХ ПРОЄКТІВ ЗА НЕЛІНІЙНОЮ РЕГРЕСІЙНОЮ МОДЕЛЛЮ

Приходько С.Б., д.т.н., професор¹; Пухалевич А.В., к.т.н.²; Машкін О.О.³

^{1,2,3} Національний університет кораблебудування імені адмірала Макарова

^{1,2,3} Україна, Миколаїв

¹sergiy.prykhodko@nuos.edu.ua; ²andrii.pukhalevych@nuos.edu.ua; ³201245@nuos.edu.ua

Анотація. В роботі виконано математичне моделювання трудомісткості розробки програмних проєктів за нелінійною регресійною моделлю в залежності від кількості функціональних точок.

Ключові слова: математичне моделювання; трудомісткість; програмний проєкт; нелінійна регресійна модель.

Вступна частина. У наші дні розробка програмного забезпечення (ПЗ) залишається бути складним процесом, який не завжди є успішним. Так, за дослідженнями Standish Group, які виконувалися з 2016 по 2020 р., лише 31% програмних проєктів були успішними, 50% завершилися лише з частковим успіхом, а 19% не були завершені взагалі [1], що у порівнянні з попередніми результатами, які були представлені у 2015 р. [2], практично не змінилися.

Однією із причин цього є не достовірне прогнозування трудомісткості розробки програмних проєктів в певних випадках. Тому достовірне прогнозування трудомісткості розробки програмних проєктів залишається важливою задачею.

На сьогоднішній день при прогнозуванні трудомісткості розробки програмних проєктів часто використовуються нелінійні регресійні рівняння трудомісткості їх виконання в залежності від певних метрик ПЗ, наприклад, таких як кількість строк коду, функціональних точок, сторінок проєктної документації, дефектів, тощо. Але зазначені рівняння, зокрема СОСОМО 81, СОСОМО II [3], не містять у своєму складі випадкових складових, що не дозволяє здійснювати моделювання трудомісткості розробки програмних проєктів як випадкової величини. Тому виникає потреба у побудові відповідних моделей для моделювання трудомісткості розробки програмних проєктів саме як випадкової величини.

Метою дослідження є побудова нелінійної регресійної моделі для моделювання трудомісткості розробки програмних проєктів саме як випадкової величини.

Основна частина. Розглянемо побудову нелінійної регресійної моделі для моделювання трудомісткості розробки програмних проєктів саме як випадкової величини на основі нормалізуючого перетворення у формі десятикового логарифму. Як і в [4], де було побудоване нелінійне регресійне рівняння за даними з [5], ми брали такі метрики з 133 програмних проєктів: трудомісткість Y у годинах; кількість функціональних точок X_1 . Далі ми перевірили двовимірні дані на нормальність за допомогою тесту Мардіа (Mardia's test), якій 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 за десятиковим логарифмом даних.

Використовуючи зазначені вище дані з 133 програмних проєктів спочатку була побудована лінійна регресійна модель $Y = \hat{b}_0 + \hat{b}_1 X_1 + \varepsilon$, де $\hat{b}_0$ та $\hat{b}_1$ є оцінками параметрів.

Причому змінна ε повинна бути випадковою величиною з розподілом Гаусу. Нульова гіпотеза про нормальність розподілу ε була перевірена за критерієм Пірсона. Ця гіпотеза була відкинута для рівня значимості 0,05. Те, що розподіл ε не є гаусівським вказує на відсутність теоретичного обґрунтування застосування лінійної регресійної моделі та на необхідність побудови нелінійної регресійної моделі.

Нелінійна регресійна модель має вигляд $Y = 10^{\varepsilon + \hat{b}_0} X_1^{\hat{b}_1}$, де ε – гаусівська випадкова величина з нульовим математичним сподіванням та середньоквадратичним відхиленням, оцінка якого дорівнює 0,1208, $\hat{b}_0$ та $\hat{b}_1$ є оцінками параметрів, які мають відповідно такі значення: -0,389258 та 0,876336. Оцінки параметрів були отримані за допомогою методу найменших квадратів. Ця модель була побудована за методом [6] із врахуванням наявності викидів за даними з 95 програмних проєктів, які залишилися після видалення викидів.

Створена нелінійна регресійна модель була перевірена за множинним коефіцієнтом детермінації R^2 , середньою величиною відносної помилки MMRE і відсотком прогнозованих результатів, для яких величини відносної помилки MRE менші за 0,25, PRED(0,25). Допустимі значення MMRE і PRED(0,25) складають не більше 0,25 і не менше 0,75 відповідно. Значення R^2 , MMRE і PRED(0,25) для створеної моделі дорівнюють відповідно 0,875, 0,230 і 0,621. Це свідчить про задовільну якість отриманої моделі.

Висновки. В роботі на основі десятичного логарифму була побудована нелінійна регресійна модель для моделювання трудомісткості розробки програмних проєктів саме як випадкової величини в залежності від кількості функціональних точок.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Portman, H. (2021, January 6). *Review Standish Group – CHAOS 2020: Beyond Infinity*. <https://hennyporportman.wordpress.com/2021/01/06/review-standish-group-chaos-2020-beyond-infinity/>
- [2] Wojewoda, S., & Hastie, S. (2015, October 4). *Standish Group 2015 Chaos Report – Q&A with Jennifer Lynch*. Standish Group. <https://www.infoq.com/articles/standish-chaos-2015/>
- [3] Boehm, B.W., Abts, C., Brown, A.W., Chulani, S., Clark, B.K., Horowitz, E., Madachi, R., Reifer, D., & Steece, B. (2000). *Software Cost Estimation With Cocomo II*. Prentice Hall.
- [4] Prykhodko, S.B., Prykhodko, N.V., Makarova, L.M., Kudin, O.O., & Smykodub, T.G. (2017). Constructing the non-linear regression equations on the basis of bivariate normalizing transformations. *Вісник Херсонського національного технічного університету*. 3(62). 333-337.
- [5] Kitchenham, B., Pfleeger, S.L., McColl, B., & Eagan, S. (2002) An empirical study of maintenance and development estimation accuracy. *The Journal of Systems and Software*, 1(64). 57-77. [https://doi.org/10.1016/S0164-1212\(02\)00021-3](https://doi.org/10.1016/S0164-1212(02)00021-3)
- [6] Prykhodko, S., & Prykhodko, N. (2022). A modified technique for constructing nonlinear regression models based on the multivariate normalizing transformations. *Selected Papers of the VIII International Scientific Conference "Information Technology and Implementation" (IT&I-2021). Workshop Proceedings (IT&I-WS-2021), Kyiv, Ukraine, December 1-3, 2021. CEUR Workshop Proceedings*, 3179, 156-166. http://ceur-ws.org/Vol-3179/Paper_15.pdf

Prykhodko S.B., Pukhalevych A.V., Mashkin O.O.

Mathematical modeling of software project development effort by a nonlinear regression model

Abstract. In work mathematical modeling of software project development effort by a nonlinear regression model depending count of functional points has been performed.

Keywords: mathematical modeling; effort; software project; nonlinear regression model.

УДК 004.412:519.237.5

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ТРУДОМІСТКОСТІ AGILE ТЕСТУВАННЯ НЕВЕЛИКИХ ВЕБ ПРОЄКТІВ ЗА НЕЛІНІЙНОЮ РЕГРЕСІЙНОЮ МОДЕЛЛЮ

Приходько С.Б., д.т.н., професор¹; Пухалевич А.В., к.т.н.²; Сорокіна Є.В.³

^{1,2,3} Національний університет кораблебудування імені адмірала Макарова
^{1,2,3} Україна, Миколаїв

¹sergiy.prykhodko@nuos.edu.ua; ²andrii.pukhalevych@nuos.edu.ua; ³211266@nuos.edu.ua

***Анотація.** В роботі виконано математичне моделювання трудомісткості agile тестування невеликих веб проєктів за трифакторною нелінійною регресійною моделлю.*

***Ключові слова:** математичне моделювання; трудомісткість; тестування; веб проєкт; нелінійна регресійна модель.*

Вступна частина. Прогнозування трудомісткості тестування програмного забезпечення (ПЗ) є однією з важливих задач розробки ПЗ. Agile тестування — це процес тестування ПЗ, який відповідає принципам гнучкої розробки ПЗ [1]. У порівнянні з каскадним тестуванням, це відносно новий підхід до тестування ПЗ, який призводить до зменшення трудомісткості тестування. Agile тестування добре підходить для невеликих програмних проєктів, включаючи невеликі веб проєкти.

Життєвий цикл agile тестування складається з 5 етапів [2], другим з яких є планування agile тестування, яке включає прогнозування трудомісткості тестування. У свою чергу прогнозування трудомісткості тестування є складною задачею, для вирішення якої застосовуються різноманітні математичні моделі, включаючи і регресійні [3, 4]. Зазначимо, що регресійні моделі містять у своєму складі випадкові складові, що дозволяє здійснювати моделювання трудомісткості тестування програмних проєктів як випадкової величини. В [4] було побудовано декілька нелінійних регресійних моделей для оцінювання трудомісткості agile тестування невеликих веб проєктів, але не були наведені значення оцінок середньоквадратичного відхилення випадкової складової, яка описує залишки. А це не дозволяє здійснювати моделювання трудомісткості agile тестування. Тому виникає потреба в удосконаленні зазначених моделей для моделювання трудомісткості agile тестування невеликих веб проєктів саме як випадкової величини.

Метою дослідження є удосконалення нелінійної регресійної моделі для моделювання трудомісткості agile тестування невеликих веб проєктів саме як випадкової величини.

Основна частина. Розглянемо побудову нелінійної регресійної моделі для моделювання трудомісткості agile тестування невеликих веб проєктів саме як випадкової величини на основі нормалізуючого перетворення у формі десятикового логарифму. Як і в [4], ми брали такі метрики з 40 проєктів: трудомісткість Y у годинах; кількість тестових випадків X_1 ; кількість сторінок програмно-технічної документації X_2 ; кількість дефектів X_3 . Далі ми перевірили чотиривимірні дані на нормальність за допомогою тесту Мардіа (Mardia's test), 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 за десятиковим логарифмом даних зазначених вище метрик. Викидів не було.

Використовуючи зазначені вище дані з 40 проєктів спочатку була побудована лінійна регресійна модель $Y = \hat{b}_0 + \hat{b}_1 X_1 + \hat{b}_2 X_2 + \hat{b}_3 X_3 + \varepsilon$, де $\hat{b}_0$, $\hat{b}_1$, $\hat{b}_2$ та $\hat{b}_3$ є оцінками параметрів. Причому змінна ε повинна бути випадковою величиною з розподілом Гаусу. Нульова гіпотеза про нормальність розподілу ε була перевірена за критерієм Пірсона. Ця гіпотеза була відкинута для рівня значимості 0,05. Те, що розподіл ε не є гаусівським вказує на відсутність теоретичного обґрунтування застосування лінійної регресійної моделі та на необхідність побудови нелінійної регресійної моделі.

Нелінійна регресійна модель має вигляд $Y = 10^{\varepsilon + \hat{b}_0} X_1^{\hat{b}_1} X_2^{\hat{b}_2} X_3^{\hat{b}_3}$, де ε – гаусівська випадкова величина з нульовим математичним сподіванням та середньоквадратичним відхиленням, оцінка якого дорівнює 0,0294, $\hat{b}_0$, $\hat{b}_1$, $\hat{b}_2$ та $\hat{b}_3$ є оцінками параметрів, які мають відповідно такі значення: -0,45001, 0,78887, 0,00176 та 0,18782. Оцінки параметрів були отримані за допомогою методу найменших квадратів. Ця модель була побудована за методом [5] із врахуванням наявності викидів за даними з 40 програмних проєктів.

Створена нелінійна регресійна модель була перевірена за множинним коефіцієнтом детермінації R^2 , середньою величиною відносної помилки MMRE і відсотком прогнозованих результатів, для яких величини відносної помилки MRE менші за 0,25, PRED(0,25). Допустимі значення MMRE і PRED(0,25) складають не більше 0,25 і не менше 0,75 відповідно. Значення R^2 , MMRE і PRED(0,25) для створеної моделі дорівнюють відповідно 0,981, 0,0503 і 1,000. Це свідчить про добру якість отриманої моделі.

Висновки. В роботі на основі десятичного логарифму була побудована нелінійна регресійна модель для моделювання трудомісткості agile тестування невеликих веб проєктів саме як випадкової величини в залежності від трьох факторів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Penmetsa, J.R. (2017). Agile Testing. In H. Mohanty, J. Mohanty, & A. Balakrishnan (Eds), *Trends in Software Testing* (pp. 19-33). Springer, Singapore. https://doi.org/10.1007/978-981-10-1415-4_2
- [2] ReQtest. (2018, July 18). *Agile Testing – Principles, methods & advantages*. <https://reqtest.com/testing-blog/agile-testing-principles-methods-advantages/>
- [3] Lazić, L., Đokić, I., & Milinković, S. (2014). Challenges in estimating software testing effort. *Proceedings of INFOTEH-JAHORINA, Vol.13*. 637-642.
- [4] Prykhodko N.V., & Prykhodko, S.B. (2019). A multiple non-linear regression model to estimate the agile testing efforts for small Web projects. *Radio Electronics, Computer Science, Control*, 2(49). 158-166. <https://doi.org/10.15588/1607-3274-2019-2-17>
- [5] Prykhodko, S., & Prykhodko, N. (2022). A modified technique for constructing nonlinear regression models based on the multivariate normalizing transformations. *Selected Papers of the VIII International Scientific Conference "Information Technology and Implementation" (IT&I-2021). Workshop Proceedings (IT&I-WS-2021), Kyiv, Ukraine, December 1-3, 2021. CEUR Workshop Proceedings*, 3179, 156-166. http://ceur-ws.org/Vol-3179/Paper_15.pdf

Prykhodko S.B., Pukhalevych A.V., Sorokina Y.V.

Mathematical modeling of efforts of agile testing small web projects by a nonlinear regression model

Abstract. In work mathematical modeling of efforts of agile testing small web projects by a three-factor nonlinear regression model has been performed.

Keywords: mathematical modeling; effort; testing; web project; nonlinear regression model.

УДК 004.023:004.08

ВИКОРИСТАННЯ МЕТОДІВ РОЗКРОЮ-ПАКУВАННЯ ПРИ ВИРОБНИЦТІ ВІТРИЛ

Фаріонов Микола Володимирович

Національний університет кораблебудування імені адмірала Макарова

Україна, Миколаїв

kolia.farionov@gmail.com

Анотація. Досліджено процес виробництва вітрил та існуючі алгоритми вирішення задачі розкрою-пакування. Виявлено перспективність впровадження метавеврестичних методів на виробництві.

Ключові слова: CAD система; розкрій деталей довільної форми; метавеврестичні методи; фігурний розкрій.

Вступна частина. Сьогодні в світі дуже популярний вітрильний спорт, проводиться багато змагань, чемпіонати різних країн, Європи, Світу, також він входить в програму Олімпійських ігор. Вітрильний спорт має велику кількість шанувальників не тільки професіоналів, але й любителів активного відпочинку, яким просто подобається кататися на вихідних або ходити в походи.

Проблема економії ресурсів є актуальною та сучасною для всіх сфер людської діяльності як у минулому, так і в найближчому майбутньому. Для кожної яхти потрібні вітрила і їх виготовлення займає значну частину часу та ресурсів. Під час виробничого процесу вітрил, через напівручне розташування деталей, нераціонально використовується тканина.

Метою роботи є дослідження існуючих методів розкрою-пакування та можливості використання їх для автоматизації та зменшення відходів при виробництві вітрил.

Основна частина. Яхти поділяються на багато категорій, мають різне призначення, але є одне те, що об'єднує всіх – це вітрила. Зараз у світі поширені 3 типи розкроїв, вони залишились як найефективніші зі сторони повторення ідеальної форми змодельованого вітрила.

Моделювання вітрила починається з замірів необхідних величин. Вітрило - це об'ємна фігура і для того щоб із плоскої тканини отримати об'ємне вітрило треба використовувати спеціальний набір плоских елементів. За допомогою спеціальної CAD системи та необхідних розмірів розробляється шаблон-розкладка. Шаблон – це набір геометричних планіметричних фігур, за допомогою яких можна представити об'ємне вітрило розкладене на площині. Набір фігур складається з різноманітних трапецій та трикутників, сторони яких можуть бути як увігнуті так і опуклі. Основні типи розкроїв:

1. Радіальний – панелі мають вид трикутників та трапецій;
2. Паралельний – панелі тільки трапецієвидної форми паралельні одна одній і перпендикулярні до задньої шкаторини;
3. Комбінований – окремі частини вітрила виконують по різним технологіям.

Існує багато виробників тканини для вітрильного спорту, так само існує багато типів тканин. Важкі та жорсткі дакронові тканини для лавірувальних вітрил, легкі та міцні нейлонові та поліестрові тканини для спінакерів та генакерів, легкі тканини з плівки. Більшість сучасних тканин мають однаковий натяг по утіку та по основі волокон, тому можливо робити різні в будь-яких напрямках і відповідно розташовувати панелі на рулоні у

хаотичному порядку. Так само існують тканини, по більшій мірі пливчасті, для яких розташування панелей має бути строго вздовж полотна, тому що натяг волокон відрізняється і розріз під кутом знищить переваги різного натягу та у найгіршому випадку зруйнує структуру тканини, що призведе до втрати промислових характеристик і швидкому руйнуванню вітрила.

Розкладка деталей вітрила відбувається на рулонах тканини стандартної ширини 150см, 140 см або 90 см. Довжина рулону тканини може варіюватися тому що, виробник тканини відбраковує деякі частини виробленого матеріалу і таким чином рулон тканини ділиться на декілька не рівномірних частин. Так як рулон тканини це прямокутник для розрахунків введемо позначення w – довжина та h – ширина рулону. Система різання матеріалу дозволяє робити розрізи у будь яких напрямках. Використовуючи лазерний плотер відстань між деталями краще обирати щонайменше 5мм, а відстань від країв рулону тканини – 10мм.

Маємо рулон тканини довжини w та ширини h . Повинні розмістити панелі з набору для вітрила на ньому так, щоб максимально ефективно використати його площину, тобто мінімізувати втрати тканини. Набір для вітрила складається з n панелей трикутної та, або тільки трапецевидної форми. В залежності від типу тканини маємо або не маємо можливість повертати панелі на кут α . Розміщення деталей проводиться на прямокутних листах довжиною не більше 12м та висотою не більше 150 см, це обумовлено граничними характеристиками використовуваного на виробництві плотеру ФК-20. Для збільшення використання ефективної площини, для розміщення панелей можна використовувати одразу увесь рулон тканини (напівнескінчену поверхню), а вже потім розділяти його на під частини довжина яких, не може перевищувати максимальну довжину поля плотеру.

Відсоток ефективного використання площі рулону тканини можна вирахувати за формулою $\eta = \frac{S_{\text{вітрила}}}{w * h} * 100\%$;

Стандартне формулювання завдання розкрою (але не єдине) передбачає список з m замовлень (у нашому випадку полотен не більших 12 метрів), у кожному замовленні потрібно розташувати $q_j, j = 1, 2, \dots, m$ деталей. Утворимо всі можливі комбінації розкрою (“карти розкрою”) і пов’яжемо з кожною картою позитивну цілу змінну x_i , що показує, скільки разів карта використовувалася. Випишемо завдання лінійного програмування:

$$\sum_{i=1}^n c_i x_i ; \sum_{i=1}^n a_{ij} x_i \geq q_j, \forall j = 1, \dots, m; x_i \geq 0,$$

де a_{ij} – скільки разів замовлення j з’являється в карті i та c_i – ціна (часто втрати) карти i .

Точна природа обмежень може призвести до трохи різних математичних характеристик. Наведені обмеження є обмеженнями на мінімум (щонайменше задана кількість має бути зроблено, але не виключається, що буде зроблено і більше, тобто залишиться місце під деталі іншого вітрила). Якщо $c_i=1$, потрібно мінімізувати кількість шматків вихідного матеріалу, що розрізаються. Якщо обмеження з нерівностей замінити на рівність, завдання називається упаковкою в контейнери.

Таке формулювання завдання можна застосувати не тільки для одновимірного випадку. Можливо багато варіацій, наприклад, мета – не мінімізація втрат, а максимізація загальної кількості виробленого товару, тобто розташувати якомога більшу кількість панелей. У цілому, кількість можливих карт розкрою зростає експоненційно від m , числа замовлень. При зростанні числа замовлень може виявитися непрактичним перераховувати можливі карти розкрою.

Як альтернативу можна використовувати генерацію стовпів. Цей метод вирішує завдання розкрою, починаючи з кількох карт. Метод утворює нові карти, якщо вони будуть

потрібні, у процесі рішення. Метод відкладеної генерації стовпів може виявитися набагато ефективнішим за вихідний підхід, особливо при зростанні розміру завдання. Відкладена генерація стовпів щодо застосування розкрою вперше використана Гілмором і Гоморі в серії статей, опублікованих у 1960-х роках[1][2]. Вони показали, що цей підхід наводить гарантовано до (дрібного) оптимального рішення без необхідності перераховувати всі можливі карти заздалегідь.

Вихідний метод Гілмору і Гоморі не був цілочисленним, так що рішення могло містити дробові складові, наприклад, деяка карта мала використовуватися 3,67 разів. Округлення до найближчого цілого часто не працює, в тому сенсі, що це призводить до не повністю оптимального рішення та недовиробництва (яке нам не підходить, бо не зможемо скласти вітрило) або надвиробництва для деяких замовлень (це не зовсім недолік для стандартизованих моно класів яхт, які використовують ідентичні вітрила, але треба вести десь підрахунок зайвих панелей, щоб не виробляти їх наступного разу). Ці недоліки подолані у нових алгоритмах, які дозволяють знаходити оптимальні рішення (у сенсі знаходження рішення з мінімальними відходами) дуже великих завдань (загалом великих, ніж потрібно на практиці [3] [4]).

У загальному випадку виділяють одновимірні, двовимірні та тривимірні завдання розкрою пакування. Задача розміщення панелей вітрил відноситься до двовимірних-фігурних задач розкрою, так як саме для цього підвиду характерні геометричні об'єкти складної опуклої, увігнутої та опукло-увігнутої форми, які не дуже відрізняються.

Вирішення завдання фігурного нерегулярного розкрою точними методами практично неможливе, як через складність самих об'єктів, так і в складності досягнення точного результату. При спробі отримання точного розв'язання таких завдань доводиться використовувати складний математичний апарат, проте, необхідна точність рішення може бути не досягнуто.

Для вирішення проблеми нерегулярного фігурного розкрою досить часто використовують евристичні методи. В разі застосування евристичного методу на одиниці операції рішення оперують геометричними перетвореннями кожного окремого геометричного об'єкту. Серед таких методів найбільшою популярністю користуються імітація відпалу, генетичний алгоритм, пошук із заборонами. Імітація відпалу та пошук із заборонами характеризуються «прокладанням» маршруту у просторі пошуку, який є послідовністю рішень, де кожне рішення є сусіднім для попереднього щодо деякої околиці. У генетичному алгоритмі перебування чергового нового рішення використовуються процедури схрещування і цілеспрямованих мутацій (змін, що відбуваються випадковим чином).

У роботі [5] – запропоновано до вирішення задачі нерегулярного розкрою плоских геометричних об'єктів використовувати метод послідовного уточнення оцінок. Застосування задачі нерегулярної упаковки передбачає використання процедури формування пріоритетного списку геометричних об'єктів за «жадібним» принципом замість жорстко зафіксованого списку.

Для вирішення задач нерегулярного розміщення також запропоновано алгоритм методу комбінаторної оптимізації. Він є ітераційним процесом, в якому кожна ітерація складається із стадії конструювання можливого рішення та стадії локального пошуку на околицях побудованого рішення. Ідея алгоритму локального пошуку полягає в наступному: подібні рішення часто поводяться подібним чином, тому навіть незначні модифікації рішень зазвичай тягнуть невеликі і зрозумілі зміни в їх якості, що дозволяє отримати кращі рішення.

Існує гібридний алгоритм оптимального розміщення панелей, що поєднує в собі ідеї мурашиних колоній та «жадібною» стратегією. Основною ідеєю алгоритму є поєднання фігур за прямими лініями. Ідеї методу мурашиної колонії засновані на зборі статичної інформації про найвдаліші знайдені рішення. При вирішенні задач нерегулярного фігурного розкрою поряд із загальновідомою апроксимацією прямокутниками пропонується об'єднання фігурних об'єктів у блоки або шматки, де під блоком розуміється набір деталей, положення яких зафіксовані щодо один одного. Даний метод відноситься до складних багатопрохідних евристик. Ідея даного алгоритму полягає в стикуванні двох деталей (опуклої і увігнутої) лише по трьом послідовним ребрах, при цьому деталь апроксимується багатокутником. Далі методом перебору вибираються три «кращі» ребра, якщо відбувся перетин деталей, то виконується процедура зсуву у напрямку бісектриси одного з кутів. Алгоритм вирішує проблему фіксованої кількості положень та орієнтацій панелей на заданій площині.

Найбільш перспективним для вирішення задачі нерегулярного розміщення геометричних об'єктів є адаптація метаевристичних методів локального пошуку в комбінації з методами моделювання геометричних перетворень, що ґрунтуються на виконанні умов взаємного неперетину. Маловивченим метаевристичним методом щільного розміщення геометричних об'єктів в однозв'язній та багатозв'язній областях є використання сплайнів та примітивів для аналізу локальних характеристик форми.

Висновки. У сучасних умовах, коли все більш актуальною стає проблема економії ресурсів на виробництві, її вирішення відбувається за всіма можливими напрямками науки, техніки та виробництва. До одного із них належить завдання нерегулярного розкрою заготовок. Вона поєднує в собі як можливість економії матеріалу, що розкроюється, так і економію енергії і часу, що витрачаються на процесі різки. Оскільки дані завдання відносяться до класу NP - важких, ускладнених геометричними особливостями об'єктів, що розкриваються, то застосування сучасних метаевристичних методів пошуку для їх вирішення є кращим рішенням.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Gilmore, P.C., & Gomory, R.E. (1961). A linear programming approach to the cutting-stock problem. *Operations Research*, issue 9 (pp. 154-162).
- [2] Gilmore, P.C., & Gomory, R.E. (1961). A linear programming approach to the cutting-stock problem, Part II. *Operations Research*, issue 11 (pp. 863-888).
- [3] Goulimis, C. (1990). Optimal solutions for the cutting stock problem. *European Journal of Operational Research*, issue 44 (pp. 197-208).
- [4] de Carvalho, V. (1998). Exact solution of cutting stock problems using column generation and branch-and-bound. *International Transactions in Operational Research*, issue 5 (pp. 35-44).
- [5] Петренко, А.В. (2005). Оптимізація розміщення двовимірних геометричних об'єктів на анізотропному матеріалі з використанням методів математичного програмування. *Міжнародна конференція ICT*.

Farionov Mykola Volodimirovich

The use of cutting-packing methods in the production of sails

Abstract. The sail production process and existing algorithms for solving the cutting-packing problem were investigated. The perspective of the implementation of metaheuristic methods in production has been revealed..

Keywords: CAD system; true type shape nesting; metaheuristic methods; figure cut.

УДК 004.412:519:237.5

РЕГРЕСІЙНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ РОЗМІРУ ВЕБ-ЗАСТОСУНКІВ, ЩО СТВОРЮЮТЬСЯ МОВОЮ PYTHON З ВИКОРИСТАННЯМ DJANGO REST FRAMEWORK

Давлатова Д.Х.¹, Макарова Л.М., к.т.н., доцент²^{1,2} Національний університет кораблебудування імені адмірала Макарова^{1,2} Україна, Миколаїв¹ didi77777@ukr.net; ² lidiia.makarova@nuos.edu.ua

Анотація. Розглянуто проблему оцінювання розміру веб-застосунків, що створюються мовою Python з використанням Django Rest Framework. Для побудови нелінійної регресійної моделі в якості нормалізуючого перетворення було використано десятковий логарифм та дані з 71 проекту з відкритим вихідним кодом, завдяки чому було підвищено достовірність оцінювання розміру відповідних веб-застосунків.

Ключові слова: нелінійна регресійна модель; веб-застосунки; логарифмічне перетворення; Python; Django Rest Framework.

Вступна частина. Успіх або невдача проекту на ранньому етапі розробки, залежить безпосередньо від ефективності оцінювання трудомісткості, яка в свою чергу залежить від розміру програми, тому одним з тих факторів, що впливають на процес розробки ПЗ, є розмір програми. Проблема отримання ефективної системи оцінювання розміру веб-застосунків – це важливе завдання, що вимагає удосконалення існуючих методів. [1].

Важливим етапом аналізу складної системи є побудова математичної моделі, що здійснюється на підставі змістовної моделі і складається з вхідних параметрів, внутрішніх параметрів та вихідних параметрів. Однією з умов побудови якісної адекватної моделі є належність вхідних даних, тому потрібна первинна обробка експериментальних даних із метою виявлення ненадійних вимірів [2]. Виявлення аномальних значень відноситься до проблеми пошуку зразків у даних, які не відповідають очікуваній поведінці.

Найбільш часто використовуваний підхід до оцінювання заснований на регресії, де модель СОСОМО є найвідомішою моделлю [3]. Багато дослідницьких робіт з оцінювання веб-зусиль також були проведені Мендесом та його співробітниками [4].

Під час проведення регресійного аналізу дослідники, зазвичай, віддають перевагу побудові лінійних моделей, які легко оцінюються, і результати яких чітко інтерпретуються. Однак дані рідко носять абсолютно лінійний характер, тому для їх адекватного опису доводиться вдаватися до побудови нелінійних залежностей.

TIOBE Software представила рейтинг найпопулярніших мов програмування на серпень 2022 року [5]. Порівняно з минулим роком Python додав до популярності 3,56%, перемістившись з другого на перше місце з показником 15,42%. Це найвищий показник популярності цієї мови програмування за час існування рейтингу. Найнижчий був зафіксований у 2003 році (0,97%), коли Python посідав 13 місце в рейтингу [6].

Існують моделі оцінювання розміру різноманітних програмних застосунків створених мовою Java, додатків реалізованих на JavaScript та веб-застосунків створених мовою PHP як із використанням фреймворків, наприклад Laravel, так і без них [6-8]. Але враховуючи те, що всі моделі безпосередньо залежать від мови програмування та її особливостей, використавши розроблені моделі для однієї мови, будуть отримані результати, що суттєво відрізнятимуться від тих, що були б отримані з використанням відповідної моделі. Тобто маємо, що для

обраної мови програмування, фреймворку та бібліотеки не було створено відповідної моделі, реалізованої для особливостей мови Python, тому є доцільним створити таку модель.

Таким чином, побудова нелінійних регресійних моделей із використанням нормалізуючих перетворень для підвищення достовірності оцінювання розміру веб-застосунків, що створюються мовою Python з використанням Django Rest Framework, і створення на їх основі інформаційної технології переробки інформації є актуальною та має практичну цінність.

Метою дослідження є підвищення достовірності оцінювання розміру веб-застосунків, що створюються мовою Python з використанням Django Rest Framework.

Основна частина. Для побудови двофакторної нелінійної регресійної моделі для оцінювання розміру веб-застосунків, що створюються мовою Python з використанням Django Rest Framework було обрано наступні фактори:

- загальна кількість рядків коду проекту в тисячах (kilolines of code, KLOC), в якості значення залежної змінної Y ;
- кількість класів проекту, в якості першої незалежної змінної X_1 ;
- кількість методів проекту, в якості другої незалежної змінної X_2 .

Практика перетворення даних, щоб вони виглядали нормально за допомогою певного нормалізуючого перетворення широко використовується через її простоту. Хоча, можливо, це не найелегантніше рішення проблеми, часто ця техніка працює добре як швидке та прагматичне рішення. Тож було обрано десятковий логарифм в якості нормалізуючого перетворення для побудови регресійної моделі для оцінювання розміру веб-застосунків, що створюються мовою Python з використанням Django Rest Framework. У логарифмічному перетворенні кожна змінна x замінюється на $\lg(x)$ з основою 10 (1) і зворотним до нього є перетворення (2):

$$z = \lg(x), \quad (1)$$

$$x = 10^z. \quad (2)$$

Багатофакторна лінійна регресійна модель в загальному випадку має вигляд:

$$Z_y = \widehat{Z_y} + \varepsilon = b_0 + b_1 Z_{x1} + b_2 Z_{x2} + \dots + b_k Z_{xk} + \varepsilon, \quad (3)$$

де: ε – це випадкова похибка, розподіл якої в загальному випадку залежить від незалежних змінних, але математичне очікування якої рівне нулю.

Нелінійна регресійна модель будується за перетвореним у вигляді десяткового логарифму та лінійним рівнянням (3):

$$Y = 10^{\varepsilon + b_0} X_1^{b_1} X_2^{b_2} \cdot \dots \cdot X_k^{b_k}. \quad (4)$$

Щоб визначити коефіцієнти рівняння регресії b застосовують різні методи (графічний, метод середніх), проте найбільшого поширення набув метод найменших квадратів (МНК). Суть МНК полягає в мінімізації суми квадратів відхилень значень пояснюваної змінної, що спостерігаються, від її розрахункових значень. За допомогою МНК було знайдено коефіцієнти $b_0 = -1,6796$, $b_1 = 0,0534$, $b_2 = 0,9571$ та з використанням формули (3) побудовано лінійну регресійну модель для нормалізованих даних:

$$Z_y = -1,6796 + 0,0534 Z_{x1} + 0,9571 Z_{x2} + \varepsilon.$$

Оскільки було виконано побудову лінійної регресії для нормалізованих даних необхідно впевнитись, що залишки лінійної регресійної моделі підкорюються нормальному закону розподілу. Перевірку залишків на нормальність також проводимо на основі критерію Пірсона.

Розрахували критичне та розрахункове значення критерію Пірсона: $\chi_{кр}^2 = 9,49$ та $\chi^2 = 0,22$ відповідно, з чого можемо зробити висновок, що залишки розподілені за нормальним законом, тобто можемо будувати дану лінійну регресію і за рівнянням (4) можемо перейти до нелінійної регресійної моделі:

$$Y = 10^{\varepsilon - 1,6796X_1^{0,0534}X_2^{0,9571}}.$$

Для нелінійної моделі було отримано значення коефіцієнту детермінації $R^2 = 0,8936$, середньої величини відносної похибки $MMRE = 0,2342$ і рівню прогнозування $PRED(0,25) = 0,6290$.

Висновки. Аналізуючи параметри якості двофакторної нелінійної регресійної моделі дійшли висновку, що можливо логарифмічне перетворення дає не найкращі результати і для нормалізації рядів даних можна було б використати перетворення Бокса-Кокса, перетворення Джонсона або багатовимірні перетворення та незважаючи на те, що значення рівня прогнозування $PRED(0,25)$ не дістало до необхідного мінімально допустимого значення 0,75, загалом по іншим параметрам було отримано гарні результати і отримана нелінійна регресійна модель має високу якість оцінювання.

Отже, двофакторну нелінійну регресійну модель, що розроблена на основі обраних незалежних змінних, а саме значення кількості класів X_1 та кількості методів X_2 проекту, можна використовувати для виконання прогнозу значень результативного показника розміру веб-застосунків у KLOC, тому на підставі саме цієї моделі можемо виконувати розробку програми для прогнозування.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Макарова, Л.М., Приходько Н.В., & Кудін О.О. (2019) Побудова нелінійної регресійної моделі для оцінювання розміру веб-додатків, реалізованих мовою Java. Вісник Херсонського національного технічного університету, № 2(69), 145–153.
- [2] Павленко, П.М. (2013). Основи математичного моделювання систем і процесів : навч. посіб. Київ: Книжкове вид-во НАУ.
- [3] Boehm, B.W. (1981). Software engineering economics. New Jersey : Prentice Hall.
- [4] Kitchenham, B., & Mendes, E. (2009). Why Comparative Effort Prediction Studies may be Invalid. International Conference on Predictor Models in Software Engineering, 1-5.
- [5] The software quality company TIOBE. (n.d.). TIOBE Index for September 2022. Retrieved September 30, 2022, from <https://www.tiobe.com/tiobe-index/>.
- [6] The software quality company TIOBE. (n.d.). The Python Programming Language. Retrieved September 30, 2022, from <https://www.tiobe.com/tiobe-index/python/>.

Davlatova D.H., Makarova L.M.

A regression model to estimate the size of web applications created in Python using the Django Rest Framework

Abstract. The problem of estimating the size of web applications created in Python using the Django Rest Framework is considered. Decimal logarithm and data from 71 open source projects were used as a normalizing transformation to build a nonlinear regression model, which increased the reliability of the estimation of the size of the respective web applications.

Keywords: non-linear regression model; web applications; logarithmic transformation; Python; Django Rest Framework.

УДК 004.412:519:237.5

НЕЛІНІЙНА РЕГРЕСІЙНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ РОЗМІРУ ПЕРСОНАЛЬНИХ ЗАСТОСУНКІВ-ОРГАНАЙЗЕРІВ, ЩО СТВОРЮЮТЬСЯ МОВОЮ KOTLIN ДЛЯ АНДРОЇД

Телехан А. М.¹; Макарова Л.М., к.т.н, доцент²

^{1,2} Національний університет кораблебудування імені адмірала Макарова

^{1,2} Україна, Миколаїв

¹211267@nuos.edu.ua; ² lidiia.makarova@nuos.edu.ua

Анотація. Побудовано нелінійну регресійну модель для оцінювання розміру персональних застосунків-органайзерів, що створюються мовою Kotlin для Андроїд. Виконано оцінювання якості моделі, побудованої на основі емпіричних даних, зібраних з ресурсу GitHub.

Ключові слова: нелінійна регресійна модель; оцінювання розміру; Kotlin; персональні застосунки-органайзери; нормалізуюче перетворення.

Вступна частина. При розробці будь-якого програмного забезпечення пред'являється низка вимог і вирішуються задачі оцінювання, і персональні застосунки-органайзери, що створюються мовою Kotlin для Андроїд не є виключенням. Перш за все для успішної реалізації програмного забезпечення необхідні задачі оцінювання трудомісткості та ресурсів, оскільки максимально точне оцінювання необхідних трудових витрат на написання коду програмного продукту є однією з основних проблем при управлінні проектами розробки програмного забезпечення [1]. При розробці програмних проектів одного класу найбільший вплив на трудомісткість має розмір застосунків.

Аналіз сучасних методів та моделей оцінювання розміру програмного забезпечення показав, що на сьогодні більшість з математичних моделей, що дозволяють оцінити розмір програмних продуктів, стають непридатними до використання через інший клас задач, відсутність даних, ресурсів або експертних навичок в цій галузі.

У цьому дослідженні з метою оцінювання розміру програмного забезпечення було обрано метод побудови нелінійної регресійної моделі. Побудована регресійна модель дозволила підвищити достовірність та швидкість оцінювання розміру персональних застосунків-органайзерів, що створюються мовою Kotlin для Андроїд.

Метою роботи є побудова нелінійної регресійної моделі для оцінювання розміру персональних застосунків-органайзерів, що створюються мовою Kotlin для Андроїд.

Основна частина. Емпіричні дані для побудови нелінійної регресійної моделі були зібрані на інтернет-ресурсі GitHub [2], з проектів з відкритим кодом. Було відібрано 49 проектів з розробки персональних застосунків-органайзерів, що створюються мовою Kotlin для Андроїд. Для кожного проекту було зібрано метрики програмного коду, такі як кількість рядків коду програми (KLOC) та загальна кількість класів (NC).

Для досягнення поставленої мети було виконано передобробку емпіричних даних. При аналізі даних на викиди використано квадрат відстані Махаланобіса, критерій Пірсона та критерій Фішера. Елементи вибірки, що перевищують критичні значення є викидами [3]. Такі елементи було видалено з вибірки і розрахунок квадрату відстані Махаланобіса та статистик повторено. В результаті розрахунків було знайдено і видалено 5 викидів. В результаті перевірки зібраних даних на нормальність закону розподілу з використанням

критерію згоди Пірсона [4] з'ясувалося, що дані не підпорядковувалися закону Гауса - не являлися нормально розподіленими. Тому було виконано нормалізацію даних з використанням нормалізуючого перетворення на основі десяткового логарифму.

Побудовано лінійну регресійну модель для нормалізованих даних. Побудовано довірчі інтервали та інтервали передбачення для лінійної моделі. За отриманою лінійною регресійною моделлю була побудована нелінійна регресійна модель, довірчі інтервали та інтервали передбачення для моделі, із застосуванням перетворення, зворотного до нормалізуючого:

$$Y = 10^{e-1,2899} X_1^{1,0283}.$$

Для перевірки якості побудованої моделі було обчислено коефіцієнт детермінації $R^2=0,8878$, середню величину відносної похибки $MMRE = 0,2705$ та рівень прогнозування $PRED(0,25) = 0,5227$ [5]. Отримане значення R^2 є більш ніж задовільним. Значення $MMRE$ також є достатньо задовільним і знаходиться у допустимих межах ($<0,3$). Однак значення $PRED(0,25)$ є незадовільним, з чого можна зробити декілька висновків щодо покращення результатів моделювання: потрібно використовувати інший вид нормалізуючого перетворення, наприклад перетворення Джонсона, або будувати багатфакторну модель, якість котрої перевищує відповідну однофакторну модель.

Висновки. Було побудовано нелінійну регресійну модель для оцінювання розміру персональних застосунків-органайзерів, що створюються мовою Kotlin для Андроїд. Розраховано показники якості побудованої нелінійної регресійної моделі. В подальшому передбачається розробка відповідного програмного забезпечення для автоматизації процесу розрахунків і прогнозування розміру персональних застосунків-органайзерів, що створюються мовою Kotlin для Андроїд, за допомогою побудованої нелінійної регресійної моделі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Котов, С.Л., & Демирский, А.А. (2017). Информационно-аналитическая система оценивания трудозатрат и стоимости создания программных средств. Программные продукты и системы. 3 (30), 469-474.
- [2] GitHub. (n.d.). Retrieved August 19, 2022, from <https://github.com/>
- [3] Prykhodko, S., Prykhodko, N., Makarova, L., & Pugachenko, K. (2017). Detecting outliers in multivariate non-Gaussian data on the basis of normalizing transformations. 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON). doi:10.1109/ukrcon.2017.8100366
- [4] Приходько, С.Б., & Приходько, Н. В. (2018). Побудова нелінійних регресійних рівнянь на основі багатовимірних нормалізуючих перетворень. Вісник Харківського національного університету імені В.Н. Каразіна, серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління». 3 (39), 61-68.
- [5] Домбровский, В.В. (2004). Эконометрика. М.: Нов.учебник

Telekhan A.M., Makarova L.M.

Nonlinear regression model for estimating the size of personal applications created for Android using the Kotlin programming language

Abstract. A nonlinear regression model to estimate the size of personal organizer developed in Kotlin for Android was built. Evaluation of the quality of the model built based on empirical data collected from the GitHub resource was performed.

Keywords: nonlinear regression model; estimating the software size; personal organizer; Kotlin; normalizing transformation.

УДК 519.673

МАТЕМАТИЧНІ МОДЕЛІ ТА ОСОБЛИВОСТІ ЧИСЕЛЬНИХ РОЗРАХУНКІВ ПРИ РОЗРОБЦІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДОСЛІДЖЕННЯ ДИНАМІЧНИХ ВЛАСТИВОСТЕЙ ЕЛЕКТРОПРИВОДІВ

Лук'яненко П.В., магістрант

Національний університет кораблебудування імені адмірала Макарова

Україна, Миколаїв

pyotr77@gmail.com

Анотація. Робота призначена аналізу математичних моделей та особливостей чисельних розрахунків при розробці програмного забезпечення дослідження динаміки систем автоматичного керування електроприводами: реакція системи на різні види вхідного впливу, побудова логарифмічних амплітудної частотної характеристики (ЛАЧХ) та фазочастотної характеристики (ЛФЧХ), побудова діаграм Найквіста та Михайлова.

Ключові слова: ЛАЧХ, ЛФЧХ, діаграма Найквіста, годограф Михайлова, система автоматичного керування, автоматизований електропривод.

Вступна частина. Одним з основних напрямів науково-технічного прогресу в різних галузях народного господарства є механізація і автоматизація. В процесі рішення цих завдань передбачається створення і використання гнучких виробничих систем (ГВС) і робототехнічних комплексів (РТК). Це стало можливим у зв'язку із застосуванням автоматизованих електроприводів.

Вирішення питань автоматизації виробничих процесів в значній мірі залежить від рівня розвитку техніки. Широке застосування верстатів з облаштуваннями числового програмного керування (ЧПК) привело до поглиблення завдань автоматизації виробництва [1].

Для керування виконавчими облаштуваннями верстатів з ЧПК велике значення має використання регульованих електроприводів, в якості яких застосовуються комплектні електроприводи постійного і змінного струму. Основним призначенням комплектного електроприводу є його використання у верстатах в якості приводу головного руху або якості приводу подачі.

Одним з найважливіших чинників, що впливають на стабільність позиціонування обробки і на ідентичність параметрів при обробці партії деталей, є характер перехідного процесу по управляючій дії в замкнутих системах електроприводів [2].

Основна частина. Для налаштування автоматизованих електроприводів, які оптимізуються за декількома критеріями якості управління, розроблені спеціальні стандартні налаштування, що дозволяють знайти компромісне вирішення цього питання. Проте стандартні налаштування часто не задовольняють реальним вимогам виробництва, а розрахунок налаштувань під конкретні вимоги - процес складний і трудомісткий. Кращим виходом з ситуації, що створилася, є застосування обчислювальної техніки і спеціального програмного забезпечення.

Застосування обчислювальної техніки дозволить більш повно досліджувати статичні і динамічні параметри автоматизованого електроприводу і, зокрема, визначати якісно і кількісно міру впливу налаштувань регуляторів електроприводу на його характеристики. Вибір оптимальних показників якості управління можливий після математичного

модельовання приводу і перевірки отриманих результатів на стенді. Такий метод налаштування дозволяє порівняно просто і з мінімальним простоем верстатів реалізувати необхідні параметри налаштування, значно спростити роботу того, хто налагоджує верстати з ЧПК, а також істотно допомогти здобувачам освіти у вивченні реальних систем автоматичного управління електроприводом.

Існуючі системи математичного аналізу автоматизованих систем керування мають відносно високу цінову категорію (наприклад, MathCad 15 – 30600 грн, MatLab – 1265\$ - вартість однієї мережевої академічної ліцензії [3]), тому виникає питання розробки програмного забезпечення аналізу динамічних систем автоматичного керування в учбових закладах.

Для розробки алгоритмів програми, що дозволяють досліджувати динаміку електропривода, необхідно дослідити: роботу з поліномами (додавання, різниця, множення поліномів, вивід поліномів у текстовій формі та ін.); роботу з комплексними числами (арифметичні операції; мніма, речова частина, модуль, аргумент, ступінь); роботу з передаточними функціями (арифметичні операції, розрахунок ЛАЧХ, ЛФЧХ, діаграми Найквіста та Михайлова); рішення диференціальних рівнянь (розрахунок рішення диференціального рівняння методом Рунге-Кутти; диференціювання таблично заданих функцій інтерполяційним поліномом Ньютона). За допомогою даного методу розраховується реакція системи автоматичного керування (САК) на дельта-функцію, лінійний сигнал, квадратичний та гармонійний вхідні сигнали.

Початковими даними для дослідження динаміки САК є: математична модель електропривода (передаточна функція); тип задаючої дії; параметри інтегрування.

Розробку програмного забезпечення доцільно розбити на декілька етапів:

1. Розробка математичної моделі досліджуваного об'єкту;
2. Обґрунтування і вибір методу інтегрування;
3. Розробка алгоритмів рішення диференціальних рівнянь;
4. Розробка інтерфейсу програми.

Математична модель системи керування задана у вигляді передатної функції, що визначає відношення вихідного сигналу об'єкту до вхідного. У загальному випадку передатна функція має вигляд [4]:

$$W(p) = \frac{a_n \cdot p^n + a_{n-1} \cdot p^{n-1} + \dots + a_1 \cdot p + a_0}{b_m \cdot p^m + b_{m-1} \cdot p^{m-1} + \dots + b_1 \cdot p + b_0}.$$

Позначивши вхідний сигнал $Z(p)$ і вихідний сигнал $Y(p)$, отримаємо:

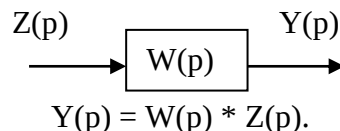


Рисунок 1 - Залежність вихідного від вхідного сигналів передатної функції

Представимо функцію $W(p)$ у вигляді двох структурних блоків, включених послідовно за схемою, зображеною на рисунку 2.

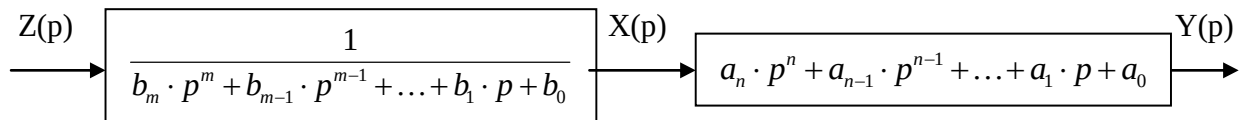


Рисунок 2 - Передатна функція у вигляді двох структурованих блоків.

При переході в часову область отримаємо:

$$b_m \frac{d^m X(t)}{dt^m} + b_{m-1} \frac{d^{m-1} X(t)}{dt^{m-1}} + \dots + b_1 \frac{dX(t)}{dt} + b_0 X(t) = Z(p),$$

$$Y(t) = a_n \frac{d^n X(t)}{dt^n} + a_{n-1} \frac{d^{n-1} X(t)}{dt^{n-1}} + \dots + a_1 \frac{dX(t)}{dt} + a_0 X(t).$$

Ввівши позначення $X' = X_1$, $X'_1 = X_2$, $\dots$, $X'_{m-2} = X_{m-1}$ отримаємо:

$$\begin{cases} X' = X_1; \\ X'_1 = X_2; \\ \dots \\ X'_{m-1} = \frac{1}{b_m} [Z(t) - b_{m-1} X_{m-1} - b_{m-1} X_{m-1} - \dots - b_1 X_1 - b_0 X] \end{cases} \quad (1)$$

$$Y = a_0 X + a_1 X_1 + \dots + a_n X_n. \quad (2)$$

Таким чином, завдання рішення диференційного рівняння зводиться до рішення системи диференціальних рівнянь (1) і отримання результату шляхом знаходження виразу (2).

Для вирішення системи рівнянь вибираємо метод Рунге-Кутти з фіксованим кроком інтегрування. Перевагою цього методу є відносно висока точність інтегрування (порядку h^4 міри, де h - крок інтегрування) при невисокій складності алгоритму.

При використанні цього методу результат рішення диференційного рівняння $y' = f(x, y)$ знаходиться по формулі:

$$Y_i = Y_{i-1} + DY_{i-1},$$

де

$$DY_{i-1} = \frac{1}{6} (K_1 + 2 \cdot K_2 + 2 \cdot K_3 + K_4),$$

$$K_1 = h \cdot f(x_{i-1}, y_{i-1}),$$

$$K_2 = h \cdot f\left(x_{i-1} + \frac{h}{2}, y_{i-1} + \frac{K_1}{2}\right),$$

$$K_3 = h \cdot f\left(x_{i-1} + \frac{h}{2}, y_{i-1} + \frac{K_2}{2}\right),$$

$$K_4 = h \cdot f(x_{i-1} + h, y_{i-1} + K_3).$$

Для аналізу стійкості системи керування необхідно розробити бібліотеки знаходження логарифмічної амплітудної характеристики, фазової характеристики, а також діаграми Найквіста та Михайлова, використовувачі задану передаточну функцію.

Висновки. Проаналізувавши існуючі системи математичного аналізу автоматизованих систем керування (MathCad, MatLab), які мають відносно високу цінову категорію, було зроблено висновок, що для реалізації програмного забезпечення, яке дозволяє досліджувати

динамічні властивості САК, доцільно розробити бібліотеки для роботи з поліномами, комплексними числами, передаточними функціями та диференціальними рівняннями.

Це надасть змогу автоматизувати та візуалізувати дослідження динаміки САК у вигляді ЛАЧХ, ЛФЧХ, діаграми Найквіста та Михайлова, а також реакції системи керування на різні види вхідного впливу.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Чернов, Е.А., & Кузьмин, В.П. (1989). *Комплектные электроприводы станков с ЧПУ: Справочное пособие*. Горький: Волго-Вятское кн. изд-во.
- [2] Терехов, В.М., & Осипов, О.И. (2006). *Системы управления электроприводов*. М.: Издательский центр «Академия».
- [3] MATLAB – это высокоуровневый язык технических расчетов, интерактивная среда разработки алгоритмов и современный инструмент анализа данных. (н.д.). <https://pro-spo.ru/information-required-to-install/749-matlab>.
- [4] Попович, М.Г., & Ковальчук, О.В. (2007). *Теорія автоматичного керування*. К:Лібідь.

Lukianenko P.

Mathematical models and features of numerical calculations in the development of software for the study of dynamic properties of electric drives

Abstract. *The work is intended for the analysis of mathematical models and features of numerical calculations in the development of software for the study of the dynamics of automatic control systems of electric drives: the response of the system to various types of input influences, the construction of logarithmic amplitude-frequency characteristics (LAFC) and phase-frequency characteristics (LPFC), construction of Nyquist and Mikhailov diagrams.*

Key words: *LAFC, LPFC, Nyquist diagram, Mikhailov's hodograph, automatic curing system, electric drive automation.*

УДК 004.412:519.237.5

МАТЕМАТИЧНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ РОЗМІРУ WORDPRESS ТЕМ НА PHP

Дорогань І.Ю.¹; Пухалевич А.В., к.т.н.²^{1,2} Національний університет кораблебудування імені адмірала Макарова^{1,2} Україна, Миколаїв¹ 211252@nuos.edu.ua; ² andrii.pukhalevych@nuos.edu.ua

Анотація. Було зібрано ООП метрики 51 Wordpress теми розроблених з використанням мови програмування PHP. Побудовано математичну модель для оцінювання розміру Wordpress тем на PHP. Перевірено характеристики побудованої моделі.

Ключові слова: оцінювання розміру програмного забезпечення; PHP; Wordpress теми; математична модель; нелінійна регресія.

Вступна частина. Оцінка розміру програмного забезпечення – це важлива інформація на перших етапах розробки, тому що оцінка розміру раніше розроблених проєктів використовується для оцінювання трудомісткості розроблення програмного забезпечення, та дозволяє правильно розподілити витрати й зусилля розробників [1]. Отже, оцінювання розміру тем для системи керування вмістом WordPress, яка є найпопулярнішою такою системою в світі [2], є важливою задачею програмної інженерії, вирішення якої потребує побудови відповідних моделей.

Для оцінювання розміру коду використовуються лінійні та нелінійні регресійні моделі [3]. Існуючі моделі, такі як трифакторна нелінійна регресійна модель для оцінювання розміру PHP застосунків із відкритим кодом [4], не призначені для оцінювання розміру WordPress тем, що призводить до низької достовірності відповідного оцінювання. Тому, розробка математичної моделі для оцінювання розміру WordPress тем на PHP є актуальною задачею.

Метою дослідження є підвищення достовірності оцінювання Wordpress тем на PHP за рахунок побудови відповідної математичної моделі.

Основна частина. Wordpress теми, розміщені на вебсайті Wordpress (<https://wordpress.org/themes>), були розроблені з використанням мови програмування PHP. ООП метрики 51 Wordpress тем з вказаного сайту були отримані за допомогою утиліти PhpMetrics (<https://phpmetrics.org/>). Для побудови математичної моделі було відібрано такі метрики: загальна кількість тисяч рядків коду Y , кількість класів X_1 , середня кількість методів на клас X_2 , глибина дерева успадкування X_3 . Вибір цього набору метрик пов'язаний з тим, що між ними відсутня мультиколінеарність.

Відібрані дані було перевірено на нормальність закону розподілу. Виявилось, що розподіл даних не являється гаусівським, тому необхідно провести нормалізацію даних. Для нормалізації емпіричних даних було використано десятковий логарифм.

Після нормалізації з даних було видалено одну з тем, так як відстань Махаланобіса перевищувала критичне значення.

Для 50 Wordpress тем, що залишились, було побудовано лінійну регресійну модель для нормалізованих значень:

$$\lg(Y) = b_0 + b_1 * \lg(X_1) + b_2 * \lg(X_2) + b_3 * \lg(X_3) + \varepsilon,$$

де b_0, b_1, b_2, b_3 – це оцінки параметрів, ε – випадкова величина.

Методом найменших квадратів було отримано наступні значення оцінок параметрів лінійної регресії $b_0 = -1,54$, $b_1 = 0,97$, $b_2 = 1,05$, $b_3 = 0,19$.

Перевірка за критерієм Пірсона показала, що розподіл випадкової величини ε є гаусівським, так як значення $\chi_2 = 6,07$ є меншим, ніж критичне значення $\chi_{2кр} = 9,49$ для 4 ступенів вільності та рівня значущості 0,05.

Нелінійна регресійна модель для оцінювання розміру Wordpress тем у тисячах рядків коду в залежності від факторів X_1 , X_2 та X_3 :

$$Y = 10^{b_0 + b_1 * \lg(X_1) + b_2 * \lg(X_2) + b_3 * \lg(X_3) + \varepsilon}.$$

Після перетворення, нелінійна регресійна модель є такою:

$$Y = 10^{\varepsilon + b_0} X_1^{b_1} X_2^{b_2} X_3^{b_3}.$$

Побудована нелінійна регресійна модель була перевірена за множинним коефіцієнтом детермінації R^2 , середньою величиною відносної помилки MMRE і відсотком оцінок кількості рядків коду, для яких величина модулю відносної помилки не більша за 0,25 (PRED(0,25)). Допустимі значення MMRE і PRED(0,25) складають не більше 0,25 і не менше 0,75 відповідно. Значення R^2 , MMRE і PRED(0,25) для побудованої моделі дорівнюють відповідно 0,92, 0,11 і 0,90. Це свідчить про високу якість отриманої моделі.

Висновки. В результаті роботи було побудовано математичну модель, для оцінювання розміру Wordpress тем на PHP. Побудована модель має високі характеристики R^2 , MMRE, PRED(0,25), що дозволяє оцінювати розмір Wordpress тем з високою достовірністю. В майбутньому запланована розробка програмного забезпечення, яке реалізує побудовану модель для оцінювання розміру Wordpress тем.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Вікіпедія. (н.д.). Оцінка складності розробки ПЗ. https://uk.wikipedia.org/wiki/Оцінка_складності_розробки_ПЗ
- [2] W3Techs. (n.d.). Usage statistics and market share of WordPress. Retrieved October 19, 2022, from <https://w3techs.com/technologies/details/cm-wordpress>
- [3] Tan, H.B.K., Zhao, Y., & Zhang, H. (2009). Conceptual data model-based software size estimation for information systems. Transactions on Software Engineering and Methodology, vol. 19, issue 2, article No. 4.
- [4] Приходько, С.Б., & Ворона, М.В. (2021). Оцінювання розміру PHP-застосунків з відкритим кодом за нелінійними регресійними моделями з різними факторами. *Збірник наукових праць Національного університету кораблебудування імені адмірала Макарова*, 1(484)/2021, 92-98. DOI: [https://doi.org/10.15589/znp2021.1\(484\).13](https://doi.org/10.15589/znp2021.1(484).13)

Dorohan I.Y., Pukhalevych A.V.

A MATHEMATICAL MODEL FOR ESTIMATING THE SIZE OF WORDPRESS THEMES IN PHP

Abstract. The OOP metrics of 51 Wordpress themes developed using the PHP programming language were collected. A mathematical model was built for estimating the size of Wordpress themes in PHP. The characteristics of the built model were checked.

Keywords: software size estimation; PHP; Wordpress themes; mathematical model; non-linear regression.

УДК 004.93

ЗАСТОСУВАННЯ КВАДРАТА ВІДСТАНІ МАХАЛАНОВІСА В ЗАДАЧІ РОЗПІЗНАВАННЯ ОБЛИЧ

Трухов А.С.¹; Приходько С.Б., д.т.н, проф.²^{1,2} Національний університет кораблебудування імені адмірала Макарова^{1,2} Україна, Миколаїв¹ timka357@gmail.com; ² sergiy.prykhodko@nuos.edu.ua

Анотація. Розроблено математичну модель для розпізнавання облич з використанням квадрата відстані Махалановіса для нормалізованих даних за допомогою перетворення Бокса-Кокса.

Ключові слова: розпізнавання образів; розпізнавання облич; математичні моделі; перетворення Бокса-Кокса; відстань Махалановіса.

Вступна частина. Методи розпізнавання образів — область технічних наук, пов'язана з розробкою принципів і побудовою систем, призначених для визначення приналежності об'єкта до певного класу. Досліджуваний об'єкт описується набором визначальних характеристик. Вибране правило прийняття рішення має ґрунтуватися на використанні обмеженої кількості характеристик, які є найбільш важливими для відмінності одного об'єкта від іншого [1]. До задач розпізнавання образів відноситься розпізнавання облич. Для вирішення цієї задачі існує багато методів, в тому числі й на основі квадрата відстані Махалановіса [2, 3]. Як зазначає ряд авторів, застосування квадрата відстані Махалановіса обумовлено нормальністю багатовимірному закону розподілу [3, 4].

Актуальність теми. Розпізнавання облич є одним із найактуальніших напрямів автоматизованої обробки інформації. Стрімкий розвиток технічного прогресу зумовив використання комп'ютерних систем та інформаційних технологій для розпізнавання облич у найрізноманітніших сферах нашого життя.

Метою роботи є побудова математичної моделі для розпізнавання облич на основі квадрата відстані Махалановіса для нормалізованих даних.

Основна частина. Кожен образ як представник відповідного класу задається за допомогою певного вектору характеристик $x = (x_1, x_2, \dots, x_n)^T$ заданим у просторі ознак X . Для вирішення задач розпізнавання образів широко застосовується класифікація на основі відстані Махалановіса [2], що обрховується за формулою (1).

$$D(x_i) = \sqrt{(x_i - \bar{x})^T S^{-1} (x_i - \bar{x})}, \quad (1)$$

де $\bar{x}$ - вибіркве середнє;

S - вибіркова коваріаційна матриця даних, яка обчислюється за формулою (2).

$$S = M[(x - M(x))(x - M(x))^T]. \quad (2)$$

Відстань Махалановіса приблизно відповідає розподілу хі-квадрат та враховує кореляцію між змінними, тобто, враховується розкид даних відносно центру мас, що має форму еліпсоїда, який описує простір допустимих значень, тому усі спостереження одного класу повинні лежати в межах цього еліпсоїду [3], в такому випадку правило прийняття рішення набуває вигляду:

$$D^2(x_i) \leq X^2(q, k), \quad (3)$$

де q – рівень значущості;

k – кількість ступенів свободи.

Формування вектору характеристик. Кожна реалізація задається вектором ознак, кожен елемент якого – нормована характеристика. В якості характеристик виступають відстані між ключовими точками обличчя. Ці точки мають бути максимально інваріантними до невеликих змін. Характеристики обираються таким чином, щоб при зміні положення, емоцій, освітлення та інших зовнішніх умов мінімально спотворювати значення вектору ознак, при цьому симетричні відстані усереднюються [5]. Для пошуку обличчя на зображенні, а також виділення ключових точок використовується ряд емпіричних алгоритмів, що виконують локалізацію особи на зображенні і потім окремих її ділянок.

За допомогою програми мовою Python та бібліотекою комп'ютерного зору Dlib для кожної фотографії набору даних [6] було сформовано вектор ознак. Dlib — система з відкритим вихідним кодом, що дозволяє безкоштовно використовувати її в будь-якій розробці. Для успішної ідентифікації обличчя може бути повернуто більш ніж на 30 градусів. Контур ока описується шістьма точками, а загальна карта обличчя описується 68 точками, що дозволяє точно порівнювати та ідентифікувати обличчя [7].

В роботі [8] автор пропонує використовувати 15 характеристик для кожного обличчя. Недоліком бібліотеки Dlib є те, що вона погано розпізнає краї обличчя, через що в залежності від повороту голови відбувається значне спотворення вектору характеристик, тому на основі цієї роботи було відібрано 11 найбільш інформативних ознак.

Задача розпізнавання значно ускладнюється зі збільшенням ознак, тому було вирішено провести розпізнавання на основі 6 характеристик: x_1 – усереднена відстань від очей до середини носа, x_2 – усереднена відстань від очей до центру рота, x_3 – усереднена відстань від очей до центру брів, x_4 – усереднена відстань від брів до верху носа, x_5 – усереднена відстань від куточків очей до верху носа, x_6 – відстань між очима. Для мінімізації впливу положення обличчя на зображенні та його віддаленості використовується нормування за допомогою ділення характеристики на базову відстань між очима [9], тому кінцевий вектор ознак має вигляд $[x_1/x_6, x_2/x_6, x_3/x_6, x_4/x_6, x_5/x_6]$.

Розпізнавання облич. Для розпізнавання було відібрано набір даних, що містить 104 фотографії однієї людини та 99 іншої.

За допомогою (3) виконується класифікація реалізацій до класу 1 або 2. При цьому кількість помилок першого роду становить 2,9%, а другого – 59,5%. Така кількість помилок розпізнавання обумовлена тим, що використання квадрата відстані Махаланобіса базується на припущенні про нормальність розподілу даних. У сфері статистики припущення про нормальність є важливим, оскільки багато статистичних методів виконують обчислення, припускаючи, що дані розподілені нормально. Обраний набір даних є ненормально розподіленими, тому виникає необхідність використання нормалізуючих перетворень.

Для підвищення ймовірності розпізнавання на основі перетворення Бокса-Кокса [10] було отримано випадкові величини, розподіл яких наближено є нормальним.

Користуючись правилом прийняття рішення (3), виконується класифікація нормалізованих реалізацій. При цьому кількість помилок першого роду становить 1,9%, а другого – 33,3%. Таким чином нормалізуюче перетворення покращило математичну модель, знизивши відсоток помилок першого роду з 2,9% до 1,9%, а другого роду з 59,9% до 33,3%.

Висновки. В ході роботи було розроблено математичну модель для розпізнавання облич, яка базується на відстані Махаланобіса. Досліджено, що математична модель, яка побудована для нормалізованих даних має більшу вірогідність правильного розпізнавання

облич. В подальшому для підвищення точності розпізнавання планується підвищити кількість ознак до 11.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Garain, U., Chaudhuri, B.B., & Ghosh, R.P. (2004). A multiple-classifier system for recognition of printed mathematical symbols. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004*. (Vol. 1, pp. 380-383). IEEE.
- [2] Fraser, A.M., Hengartner, N.W., Vixie, K.R., & Wohlberg, B.E. (2003). Incorporating invariants in Mahalanobis distance based classifiers: Application to Face Recognition. In *Proceedings of the International Joint Conference on Neural Networks, 2003*. (Vol. 4, pp. 3118-3123). IEEE.
- [3] Van Aelst, S., & Rousseeuw, P. (2009). Minimum volume ellipsoid. *Wiley Interdisciplinary Reviews: Computational Statistics*, 1(1), 71-82.
- [4] Gallego, G., Cuevas, C., Mohedano, R., & Garcia, N. (2013). On the Mahalanobis distance classification criterion for multidimensional normal distributions. *IEEE Transactions on Signal Processing*, 61(17), 4387-4396.
- [5] Starovoitov, V., Samal, D., Votsis, G., & Kollias, S. (1999). Face recognition by geometric features. In *Proceedings of 5-th Pattern Recognition and Information Analysis Conference*.
- [6] Burak. (2019). Pins Face Recognition. <https://www.kaggle.com/datasets/hereisburak/pins-face-recognition>
- [7] Boyko, N., Basystiuk, O., & Shakhovska, N. (2018). Performance evaluation and comparison of software for face recognition, based on dlib and opencv library. In *2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP)* (pp. 478-482). IEEE.
- [8] Starovoitov, V.V., Samal, D.I., & Briliuk, D.V. (2002). Three approaches for face recognition. In *The 6-th International Conference on Pattern Recognition and Image Analysis October* (pp. 21-26).
- [9] Starovoitov, V.V., & Samal, D. (1999). A geometric approach to face recognition. In *NSIP* (pp. 210-213).
- [10] Sakia, R.M. (1992). The Box-Cox transformation technique: a review. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 41(2), 169-178.

Trukhov A.S., Prykhodko S.B.

Application of the square of the mahalanobis distance in the problem of face recognition

Abstract. A mathematical model was developed for face recognition using the squared Mahalanobis distance for normalized data using the Box-Cox transformation.

Keywords: pattern recognition; face recognition; mathematical models; Box-Cox transformation; Mahalanobis distance.

УДК 519.722

АЛГОРИТМ КРИПТОГРАФІЧНОГО ПРОТОКОЛУ АВТЕНТИФІКАЦІЇ БЕЗ ПЕРЕСИЛАННЯ ПАРОЛЮ МЕРЕЖЕЮ У ЯВНОМУ ВИГЛЯДІ

Стрельников М. Д.¹; Партас В. К., к.т.н²^{1,2} Національний університет кораблебудування імені адмірала Макарова^{1,2} Україна, Миколаїв¹211227@nuos.edu.ua; ²viktor.partas@nuos.edu.ua

Анотація. У роботі приділено увагу до основних проблем мережевої безпеки при пересиланні паролів у поширених алгоритмах автентифікації. Зокрема проаналізована вразливість відкритої пересилки пароля до пасивних мережесих атак, атак типу “людина посередині” та загроза компрометації паролю, якщо він зберігається на сервері у відкритому вигляді. Розглянуто криптографічний алгоритм SPAKE2, що уникає пересилки пароля у відкритому вигляді та генерації разових сесійних спільних ключів.

Ключові слова: автентифікація; PAKE; повідомлення; пароль; криптографія.

Вступна частина. Актуальність даної роботи полягає у дослідженні способів, зменшення ризиків мережевої безпеки у протоколах автентифікації, які передбачають пересилку паролю відкритими каналами зв'язку. Патерн автентифікації із відправленням паролю публічною мережею на сервер є типовим для більшості сучасних WEB-додатків та сервісів. Велика кількість таких сервісів є дуже критичною — це можуть бути банки, ПО серверів бухгалтерії, особисті електронні кабінети, навіть соціальні мережі тощо. Подібні сценарії становлять великий рівень ризику компрометації чи втрати секретності пароля, як при активних, так і за пасивних мережесих атаках. У роботі зроблено огляд та аналіз розповсюджених векторів атаки на пароліні протоколи автентифікації, зокрема “людина посередині”. Запропоновано використовувати криптографічні протоколи, що дозволяють уникати передавання паролів у відкритому вигляді мережею .

Основна частина. Автентифікація — процедура встановлення належності користувачеві інформації в системі на основі пред'явленого ним ідентифікатора («логіна» — реєстраційного імені користувача) і пароля — певної конфіденційної інформації, знання якої передбачає володіння певним ресурсом в мережі. Протоколи авторизації за паролем реалізовані наступним чином у більшості клієнт-серверних додатків:

1. Клієнт відправляє на бік сервера логін та пароль;
2. Сервер отримує логін та пароль користувача;
3. Сервер зіставляє за базою даних логін із паролем, якщо пароль є валідний для логіна певного користувача — користувач проходить авторизацію.

Розглянемо наведену реалізацію алгоритму авторизації з точки зору мережевої безпеки. Можливі вектори атаки:

- Мережа та стек мережесих протоколів, що реалізують пересилку чутливих даних під час виконання алгоритму авторизації;
- Сервер бази даних, що зберігає ідентифікатори користувачів та резервні копії даних, СУБД;
- Серверна файлова та операційна системи, критичне системне ПО, засоби віддаленого адміністрування сервера.

В залежності від реалізації бази даних, неавторизований користувач може виконати SQL-ін'єкцію[1] (чи експлуатувати іншу вразливість СУБД) для викрадення авторизаційних даних певного користувача (чи вмісту всієї бази у гіршому випадку) та отримати неавторизований доступ до комп'ютерної системи. Потенційний зловмисник може використати вразливості конфігурації засобів віддаленого адміністрування (напр. SSH, SSH+LDAP, telnet) чи іншого критичного ПО серверу, щоб отримати доступ до файлової системи та резервних копій бази. Можна побачити, що зі складністю системи дуже швидко зростає кількість вразливих місць, а для реалізації вектору атаки буде достатньо експлуатації лише однієї вразливості. Проте, можливо набагато зменшити площу атаки на системні компоненти шляхами:

- Налаштування детального моніторингу системи, логінгу подій та проведення регулярних аудитів комп'ютерної системи та її компонентів;
- Безпечне конфігурування засобів віддаленого адміністрування, своєчасне встановлення оновлень ОС та драйверів та закриття мережевого периметру комп'ютерної системи у приватній мережі чи за VPN;
- Налаштування параметрів безпеки СУБД та БД (логінг усіх запитів, обмеження максимальної кількості під'єднань, їх тривалості, встановлення ролей користувачів БД мінімально достатніх для виконання саме їх задач та ін.);
- Шифрування файлової системи та резервних копій.

Перелік цих мір значно підвищує рівень безпеки БД у комп'ютерних системах, та підвищує загальну надійність ресурсів системи, але ускладнює комп'ютерну систему, знижує загальну швидкодію та може впливати на зручність використання. На практиці намагаються завжди відповідати компромісам, забезпечуючи і гідний рівень безпеки, і зручність. По при все, залишається фундаментально слабе місце у системі — це те, що усі авторизаційні дані зберігаються в одному місці і всі компоненти системи від нього залежать. Одночасно із цим, комп'ютерна система залишається вразливою для внутрішніх атак. Вони можуть бути, як цілеспрямованими, так і випадковими: через людський фактор та необережність, інфікування комп'ютерним вірусом обладнання, фішинг спрямований на користувачів системи чи засоби соціальної інженерії та ін. Є досить ефективне рішення: замість паролів, у базі даних зберігаються геш-значення паролів. У цьому випадку, навіть отримання неавторизованого доступу до вмісту всієї БД не призведе до катастрофічних наслідків. Але для цього потрібно слідкувати наступним рекомендаціям:

- Для формування геш-значень з паролів повинні бути використані криптографічні геш-функції, які щонайменше відповідають стандарту SHA-2 (лише SHA-256 та SHA-512, вважаються відносно надійними на сьогодні), чи краще — SHA-3 (Кессак та SHAKE);
- Для додаткового підсилення криптографічної надійності, гешування може бути виконано із додаванням “солі” — короткого рядка даних, який додається до даних на вході геш-функцій, щоб навіть за умови підбору колізії геш-функції неможливо було відновити пароль.

Але залишається загроза використання зловмисником мережевого вектора атаки (пасивної чи активної). Сценарій пасивної мережевої атаки передбачає, що конфіденційний трафік передається відкритою публічною чи домашньою мережею користувача, а потенційний зловмисник має доступ до цієї мережі. Наприклад, він може фізично під'єднатись до мережі потенційної жертви через Wi-Fi або навіть до Ethernet-кабелю за допомогою спеціального обладнання. Зловмиснику достатньо лише використати ПО, що

виконує сканування TCP-пакетів мережі (напр. Wireshark) і він отримає дані про логіни і паролі потенційних жертв у відкритому вигляді. Такі сценарії мають місце у домашніх мережах, у загальнодоступних публічних мережах чи усередині мережевого периметру певної установи. У останньому випадку, адміністратори мереж дуже рідко вмикають шифрування трафіку усередині корпоративної мережі чи багатофакторну автентифікацію, оскільки припускають, що мережа установи є цілком приватною, а доступ до інтернету відбувається лише через спеціальні шлюзи. Але це не виключає можливості того, що робочі станції користувачів корпоративної мережі будуть заражені вірусами, чи користувачі самі будуть змотивовані провести подібного роду атаку задля власної матеріальної вигоди.

Проблему пасивних мережевих атак можна досить успішно вирішити із увімкненням шифрування трафіку на різних рівнях моделі OSI[2] із використанням певного мережевого протоколу:

- WEP/WPA2/WPA3 (канальний рівень) — протоколи, які реалізують шифрування трафіку на каналному рівні у Wi-Fi мережах;
- Ipsec (мережевий рівень) дозволяє шифрувати трафік будь-якого протоколу, заснованого на IP — його часто використовують для побудови приватних мереж та VPN;
- Kerberos (протокол прикладного рівня UDP у якості транспортного протоколу) дозволяє забезпечити автентифікацію для клієнтів та серверу на рівні сокетів, а також симетричне шифрування каналу зв'язку між комп'ютерами у відкритій мережі;
- HTTPS (протокол прикладного рівня для безпечного доступу до WEB-ресурсів через браузер із TCP у якості транспортного протоколу) — розширена версія HTTP протоколу, яка додає функціонал шифрування вмісту HTTP-пакетів за допомогою сесійного ключа шифрування, узгодженого на етапі автентифікації.

Використання наведених протоколів дозволяють значно підвищити рівень безпеки мереж та майже унеможливити пасивні мережеві атаки. Проте залишається загроза активних мережевих атак, таких як “людина посередині” (англ. Man-in-the-middle)[3] та атак на мережеве обладнання. Слід зауважити, що протоколи WEP/WPA2 використовують застарілі на сьогодні стандарти шифрування (RC4), а також вразливість паролів Wi-Fi до атак прямим перебором чи атакою за словником. WPA3 на сьогодні є більш довершеним протоколом з точки зору безпеки, що протидіє також атакам прямим перебором, але ще недостатньо поширеним серед виробників мережевого обладнання. Протоколи шифрування трафіку інших рівнів моделі OSI теж мають свої вразливості. Наприклад, конфігурація протоколу Kerberos на комп'ютерах під керуванням ОС Windows за замовченням містить режим автентифікації із використанням NTLM-гешування[4], але NTLM-геші передаються відкритою мережею і є вразливим до атак із використанням “райдужних таблиць” (метод пошуку колізій геш-функцій, що використовує попередньо згенеровану множину довільних геш-функцій). Дуже поширеним є використання HTTPS — його використовують більшість WEB-додатків. Через поширеність протоколу, в ньому забезпечено зворотню взаємодію із молодшими версіями, але це є слабкою стороною протоколу. Через це існує велика кількість атак, що зосереджується саме на пониженні версії протоколу до мінімальної та подальшу експлуатацію відомих вразливостей (наприклад встановлення слабких криптографічних параметрів). Безпечнішою вважається остання версія протоколу HTTP, що включає у себе TLS1.3[5]. Безумовно, використання засобів безпеки на рівні мережевих протоколів ще значно зменшує площу потенційної атаки, але все одно не виключає такої можливості. Згідно MITRE ATT&CK (глобальна та публічна база знань задокументованих

вразливостей)[6], в HTTPS по усім існуючим версіям задокументовано близько 700 вразливостей.

Для зниження ризику розкриття паролю під час його передачі було модифіковано використання паролів у протоколах авторизації та сформульовано протоколи типу PAKE (англ. Password-Authenticated Key Exchange)[7]. Сутність PAKE полягає у тому, що сторонам заздалегідь відомо геш-значення паролю одного з учасників, тоді сторони можуть незалежно випрацювати спільний сесійний ключ. У такому випадку авторизація буде вважатися успішною, коли ключ, випрацьований сторонами за алгоритмом співпадає. Геш-значення оригінального паролю може бути сформовано при реєстрації чи за допомогою протоколу асиметричного розподілу ключів (напр. Діфі-Гелмана[8]). Серед протоколів PAKE є різні реалізації (EKE, J-PAKE, SPAKE1/2, OPAQUE та ін.). У роботі розглянуто реалізацію PAKE за стандартом SPAKE2[9], який працює на еліптичних кривих, як і більшість сучасних засобів мережевої криптографії із відкритим ключем, а саме на кривих Едвардса[10] (за стандартом Ed25519)[11]. Криві Едвардса забезпечують простішу реалізацію дій складання точок із одночасним забезпеченням високого рівня криптографічної стійкості. Крива Едвардса (рисунок 1.1) задається рівнянням $x^2 + y^2 = 1 + dx^2y^2$ (параметр $d \neq 0$ — інакше крива Едвардса стане колом).

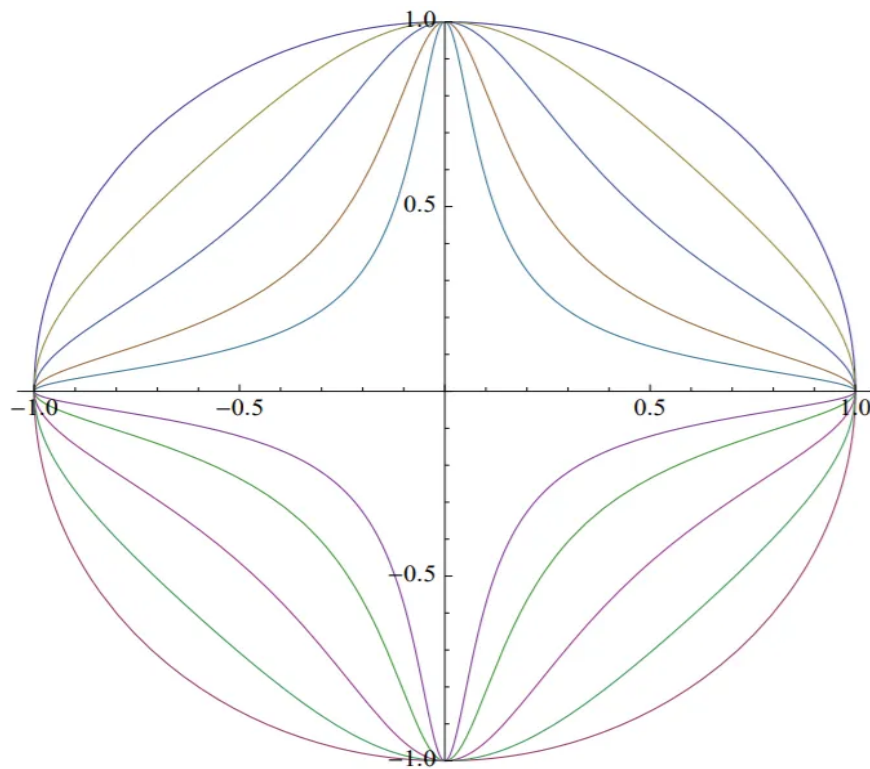


Рисунок 1.1 Сімейство кривих Едвардса .

Складання на кривих Едвардса визначається формулою (1).

$$(x_A, y_A) + (x_B, y_B) = \left(\frac{x_A y_B + x_B y_A}{1 + dx_A x_B y_A y_B}, \frac{y_A y_B - x_B x_A}{1 - dx_A x_B y_A y_B} \right). \quad (1)$$

У свою чергу, множення точки на число виконується, як багатократне складання, викрнюючі ітерації згідно формулі (1).

Параметри на вході алгоритму (відомі усім учасниками та не є секретними): публічні ключі A та B — довільні точки M та N спеціальної кривої Едвардса Ed25519; велике просте число p , G — базова точка кривої (обирається довільно)

Усі розрахунки проводяться за модулем p , а складання та множення визначені відповідно до формули (1).

Припустимо, що в нас є два абонента A та B . Тоді алгоритм виглядає наступним чином. Дії абонента A :

1. A знає спільний пароль $pass$, переводячи його до числа w за допомогою геш-функції;
2. A обирає випадковим чином число x у проміжку $[0; p]$;
3. A розраховує точки X та T згідно формули (1);
4. A відправляє точку T абоненту B .

$$\begin{aligned} w &= \text{hash}(pass) \\ x &= \text{rand}(0, p) \\ X &= xG \pmod{p} \\ T &= wM + X \pmod{p} \\ \text{Alice: } T &\rightarrow B \end{aligned} \quad (2)$$

Відповідно B діє наступним чином:

1. B отримує T ;
2. B знає спільний пароль $pass$, переводячи його до числа w за допомогою геш-функції;
3. B також обирає випадковим чином число y у проміжку $[0; p]$;
4. B розраховує Y та S згідно формули (2);
5. B відправляє Y абоненту A .

$$\begin{aligned} \text{Bob: } B &\leftarrow T \\ w &= \text{hash}(pass) \\ y &= \text{rand}(0, p) \\ Y &= yG \pmod{p} \\ S &= wN + Y \pmod{p} \\ \text{Bob: } Y &\rightarrow A \end{aligned} \quad (3)$$

Тепер A та B виконують етап верифікації незалежно один до одного:

1. A розраховує $K_B: K_B = x(S - wN) = x(wN + Y - wN) = xY = xyG \pmod{p}$;
2. B розраховує $K_A: K_A = y(T - wM) = y(wM + X - wM) = yX = xyG \pmod{p}$.

Таким чином, між сторонами A та B узгоджено ключ $K_A = K_B$, який є похідним від паролю, як наслідок пароль не передається мережею. Мережею передаються лише параметри T та Y , які не надають достатньої інформації для відновлення паролю, оскільки для зломисника залишаються невідомі x та y , що у свою чергу є секретними випадковими числами. А сам пароль $pass$ не приймає участі у розрахунках — замість нього використовується його криптографічний геш w . Тобто навіть користувачеві не потрібно зберігати оригінальний пароль $pass$ на пристрої для авторизації. За подібним алгоритмом також можна генерувати разові сесійні ключі, підсиливши рівень безпеки, шляхом зміни випадкових параметрів x та y у кожній сесії.

Висновки. Використання SPAKE2 у якості криптографічного протоколу автентифікації допомагає уникнути необхідності пересилання та зберігання паролів між клієнтом та сервером у відкритому вигляді. Корисною властивістю протоколу є те, що довжина паролю не має суттєвого значення, оскільки пароль не використовується у чистому вигляді, а тільки його відбиток. Будова протоколу SPAKE2 робить його особливо зручним у формуванні

разових сесійних ключів — це стає можливим із використанням випадкових чисел у алгоритмі генерування.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Rajbanshi, S. (2020, July 27). 3 ways of stealing data using SQL injection. <https://medium.com/devil-is-in-the-details/3-ways-of-stealing-data-using-sql-injection-71893672efcf>
- [2] Oracle. (n.d.). Open Systems Interconnection (OSI) Reference Model. <https://docs.oracle.com/cd/E19504-01/802-5886/intro-45828/index.html>
- [3] Cybercalm. (2020, May 26). Що таке атака "Man-in-the-Middle" та як себе захистити? <https://docs.oracle.com/cd/E19504-01/802-5886/intro-45828/index.html>
- [4] Microsoft MSRC. (2022, January 11). Windows Kerberos Elevation of Privilege Vulnerability (CVE-2022-21920). <https://msrc.microsoft.com/update-guide/vulnerability/CVE-2022-21920>
- [5] Sullivan, N. (2018, November 8). A Detailed Look at RFC 8446 (a.k.a. TLS 1.3). <https://blog.cloudflare.com/rfc-8446-aka-tls-1-3/>
- [6] MITRE ATT&CK. (н.д.). Globally-accessible knowledge base of adversary tactics and techniques based on real-world observations. <https://attack.mitre.org/>
- [7] Chang, C.-M., (н.д.). J-PAKE over TLS. <https://chunminchang.gitbooks.io/j-pake-over-tls/content/>
- [8] Вікіпедія. (н.д.). Протокол Діффі — Геллмана. https://uk.wikipedia.org/wiki/Протокол_Діффі_—_Геллмана
- [9] Buchanan, W. J. (2022, November 4). SPAKE2. <https://asecuritysite.com/encryption/spake>
- [10] Bernstein, D. J., Duif, N., Lange, T., Schwabe, P., Yang, B.-Y. (2017, January 22). High-speed high-security signatures. <https://ed25519.cr.yp.to/ed25519-20110926.pdf>
- [11] Housni, Y., Edwards curves. (2018, December 3). hal-01942759. <https://hal.archives-ouvertes.fr/hal-01942759/document>

Algorithm of cryptographic authentication protocol without transmitting the explicit password over network

Abstract. *The work focuses on the main problems of network security when forwarding passwords in common authentication algorithms. In particular, the vulnerability of open password transmission to passive network attacks, man-in-the-middle attacks and the threat of password compromise if it is stored on the server in open form is analyzed. The cryptographic algorithm SPAKE2, which solves the problem of sending a password in the open and generating one-time session shared keys, is considered.*

Keywords: authentication; PAKE; message; password; cryptography.

УДК 004.412:519.237.5

НЕЛІНІЙНА РЕГРЕСІЙНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ КІЛЬКОСТІ СТРОК КОДУ ВЕБ ЗАСТОСУНКІВ, ЩО СТВОРЮЮТЬСЯ ЗА ДОПОМОГОЮ ФРЕЙМВОРКУ DJUNGO

Приходько С.Б., д.т.н., професор¹; Шемчук С.Т.²; Карлюченко Є.К.³
^{1,2,3} Національний університет кораблебудування імені адмірала Макарова
^{1,2,3} Україна, Миколаїв

¹sergiy.prykhodko@nuos.edu.ua; ²sergey.shemchuk.t@gmail.com; ³211256@nuos.edu.ua

Анотація. В роботі на основі перетворення у вигляді десятичного логарифму побудована трьохфакторна нелінійна регресійна модель для оцінювання кількості строк коду веб застосунків, що створюються з використанням фреймворку Django.

Ключові слова: оцінювання кількості строк коду; веб застосунок; фреймворк Django; нелінійна регресійна модель.

Вступна частина. Оцінювання розміру веб застосунків як і іншого програмного забезпечення (ПЗ) є важливою задачею на ранній стадії розробки ПЗ, оскільки ця інформація використовується для прогнозування трудомісткості створення ПЗ за допомогою математичних моделей, наприклад, таких як СОСОМО II. Зараз багато веб застосунків створюються із використанням різноманітних фреймворків, які прискорюють розробку ПЗ. Одним із таких є Django (<https://www.djangoproject.com/>). Django – це безкоштовний високорівневий Python веб фреймворк з відкритим кодом.

На теперішній час відомі нелінійні регресійні моделі для оцінювання кількості строк коду веб застосунків з відкритим кодом в залежності від трьох метрик, що можуть бути визначені за діаграмою класів: загальна кількість класів, середня кількість методів на клас та глибина дерева спадкування [1-3]. Але ці моделі побудовані за даними веб застосунків, що створюються з використанням різних PHP-фреймворків. А Django – це на сьогодні найбільш популярний Python-фреймворк. Тому виникає потреба побудові відповідних моделей для оцінювання кількості строк коду веб застосунків, що створюються з використанням саме Python-фреймворку Django.

Метою дослідження є побудова нелінійної регресійної моделі для оцінювання кількості строк коду веб застосунків, що створюються з використанням фреймворку Django.

Основна частина. Розглянемо побудову нелінійної регресійної моделі для оцінювання кількості строк коду веб застосунків, що створюються з використанням фреймворку Django. Враховуючи на те, що Django є Python-фреймворком, метрики з 45 веб застосунків, які розміщені на сайті GitHub (<https://github.com>), були отримані за допомогою інструменту Radon (<https://radon.readthedocs.io/en/latest/index.html>) та власного скрипта (для метрик: кількість класів, методів та DIT). Як і в [3], ми брали такі метрики: розмір веб застосунку Y у тисячах рядків коду; кількість класів X_1 ; середню кількість методів на клас X_2 ; глибину дерева спадкування (DIT) X_3 . Зазначені фактори були обрані, тому що їх можна отримати з діаграми класів та між ними немає мультиколінеарності. Далі ми перевірили багатовимірні дані на нормальність за допомогою тесту Мардіа (Mardia's test), якій 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 за десятичним логарифмом даних.

Використовуючи зібрані дані із зазначених вище метрик веб застосунків була побудована лінійна регресійна модель $Y = \hat{b}_0 + \hat{b}_1 X_1 + \hat{b}_2 X_2 + \hat{b}_3 X_3 + \varepsilon$, де $\hat{b}_0$, $\hat{b}_1$, $\hat{b}_2$ та $\hat{b}_3$ є оцінками параметрів. Причому змінна ε повинна бути випадковою величиною з розподілом Гаусу. Нульова гіпотеза про нормальність розподілу ε була перевірена за критерієм Пірсона. Ця гіпотеза була відкинута для рівня значимості 0,05. Те, що розподіл ε не є гаусівським вказує на відсутність теоретичного обґрунтування застосування лінійної регресійної моделі та на необхідність побудови нелінійної регресійної моделі.

Нелінійна регресійна модель має вигляд $Y = 10^{\varepsilon + \hat{b}_0} X_1^{\hat{b}_1} X_2^{\hat{b}_2} X_3^{\hat{b}_3}$, де ε – гаусівська випадкова величина з нульовим математичним сподіванням та середньоквадратичним відхиленням, оцінка якого дорівнює 0,11109, $\hat{b}_0$, $\hat{b}_1$, $\hat{b}_2$ та $\hat{b}_3$ є оцінками параметрів, які мають відповідно такі значення: -1,26991, 0,87984, 0,72781 та 0,14788. Оцінки параметрів були отримані за допомогою методу найменших квадратів. Ця модель була побудована за методом [4] із врахуванням наявності викидів за даними з 33 застосунків, які залишилися.

Створена нелінійна регресійна модель була перевірена за множинним коефіцієнтом детермінації R^2 , середньою величиною відносної помилки MMRE і відсотком прогнозованих результатів, для яких величини відносної помилки MRE менші за 0,25, PRED(0,25). Допустимі значення MMRE і PRED(0,25) складають не більше 0,25 і не менше 0,75 відповідно. Значення R^2 , MMRE і PRED(0,25) для створеної моделі дорівнюють відповідно 0,993, 0,213 і 0,667. Це свідчить про задовільну якість отриманої моделі.

Висновки. В роботі на основі десятичного логарифму була побудована нелінійна регресійна модель з трьома предикторами для оцінювання кількості строк коду веб застосунків, що створюються з використанням фреймворку Django. Побудована модель має прийнятну якість відповідного оцінювання.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Приходько, С.Б., Приходько, Н.В., Ворона, М.В., Беловол, І.О. (2021). Нелінійна регресійна модель для оцінювання розміру Web-застосунків, що створюються з використанням фреймворку Laravel. *Інформаційні технології та комп'ютерна інженерія*, 1 (50), 115-121. DOI: <https://doi.org/10.31649/1999-9941-2021-50-1-115-121>
- [2] Prykhodko, S., Shutko, I., Prykhodko, A. (2021). Early LOC estimation of Web apps created using Yii framework by nonlinear regression models. *WSEAS Transactions on Computers*, 20, 321-328. DOI: <http://dx.doi.org/10.37394/23205.2021.20.35>
- [3] Prykhodko, S.B., Shutko, I.S., Prykhodko, A.S. (2021). A nonlinear regression model to estimate the size of Web apps created using the CakePHP framework. *Radio Electronics, Computer Science, Control*, 4 (59). 129-139. DOI: <http://dx.doi.org/10.15588/1607-3274-2021-4-12>
- [4] Prykhodko, S., Prykhodko, N. (2022). A modified technique for constructing nonlinear regression models based on the multivariate normalizing transformations. *Selected Papers of the VIII International Scientific Conference "Information Technology and Implementation" (IT&I-2021). Workshop Proceedings (IT&I-WS-2021), Kyiv, Ukraine, December 1-3, 2021. CEUR Workshop Proceedings*, 3179, 156-166. http://ceur-ws.org/Vol-3179/Paper_15.pdf

Prykhodko S.B., Shemchuk S.T., Karlyuchenko E.K.

A nonlinear regression model for estimating LOC of web apps created using the Django framework

Abstract. In work a nonlinear regression model with three predictors for estimating LOC of web applications created using the Django framework has been built.

Keywords: LOC estimation; Web app; Django; nonlinear regression model.

УДК 004.412:519.237.5

НЕЛІНІЙНА РЕГРЕСІЙНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ РОЗМІРУ СЕРВЕРНИХ ЗАСТОСУНКІВ, ЩО СТВОРЮЮТЬСЯ З ВИКОРИСТАННЯМ ФРЕЙМВОРКУ NESTJS

Латанська Л.О., к.ф.-м.н., доцент¹; Бринзей О.А.²

^{1,2} Національний університет кораблебудування імені адмірала Макарова

^{1,2} Україна, Миколаїв

¹latanskaya@gmail.com; ²oleksandrbrynzei@gmail.com

Анотація. В роботі було розроблено нелінійну регресійну модель для оцінювання розміру серверних застосунків, що створюються з використанням фреймворку NestJS.

Ключові слова: оцінка розміру програмного забезпечення; серверні застосунки; нелінійна регресійна модель.

Вступна частина. Прогнозування трудомісткості розробки програмного забезпечення є досить важливим завданням на ранніх стадіях розробки програмного забезпечення будь-якого типу, в тому числі серверних застосунків. Прогнозування трудомісткості можна проводити за допомогою таких математичних моделей, як наприклад СОСОМО II. Але в цій та інших моделях, які призначені для прогнозування трудомісткості розробки програмного забезпечення, одним із головних параметрів виступає розмір програмного забезпечення. Тому для оцінювання трудомісткості треба спочатку оцінити розмір програмного забезпечення.

На даний момент широко застосовується оточення NodeJS, з використанням якого розробляють різного роду сервери та сервіси. Для пришвидшення розробки таких застосунків використовують фреймворки, до яких відносяться: SailsJS, MeteorJS та NestJS.

Під час ознайомлення з останніми публікаціями по даній тематиці, було виявлено трьохфакторне лінійне регресійне рівняння [1], яке призначене для оцінювання кількості рядків коду програмного забезпечення, розробленого на мові PHP. Крім цього було знайдено трьохфакторну нелінійну регресійну модель [2] для оцінювання розміру PHP-застосунків з відкритим вихідним кодом, а також знайдено трьохфакторну нелінійну регресійну модель [3], яка призначена для оцінювання розміру web-застосунків, що створюються з використанням фреймворку Laravel. Із аналізу знайдених моделей можна зробити висновок, що на якість моделі впливає мова програмування, фреймворк, технології та інші фактори.

Так як не було знайдено моделі для оцінювання розміру серверних застосунків, що створюються з використанням фреймворку NestJS, то задача побудови спеціалізованої моделі для оцінювання кількості рядків коду серверних застосунків, що створюються з використанням фреймворку NestJS, є актуальною.

Метою дослідження є створення нелінійної регресійної моделі для оцінювання розміру серверних застосунків, що створюються з використанням фреймворку NestJS.

Основна частина. Розглянемо побудову нелінійної регресійної моделі для оцінювання розміру серверних застосунків, що створюються з використанням фреймворку NestJS. В якості емпіричних даних було знайдено 62 серверні застосунки, які знаходяться у відкритому доступі на сервісі GitHub (<https://github.com>). Всі вони були розроблені з використанням фреймворку NestJS, який створений і повністю підтримує TypeScript. Збір метрик проводився за допомогою інструменту SonarQube (<https://www.sonarqube.org/>). Всього було

зібрано п'ять метрик: загальна кількість рядків, кількість рядків коду, кількість файлів, кількість функцій та кількість класів. Для побудови моделі були відібрані наступні метрики: кількість рядків коду Y ; кількість функцій X_1 ; кількість класів X_2 . Вказані метрики були обрані у зв'язку з тим, що між ними відсутня мультиколінеарність. Після цього дані були перевірені на нормальність розподілу за допомогою критерія Пірсона. Тест показав, що розподіл даних не є гаусівським. Для визначення наявності багатовимірних викидів у даних, було використано тест на основі квадрату відстані Махаланобіса.

Використовуючи зібрані дані із зазначеними вище метриками, була побудована лінійна регресійна модель $Y = \hat{b}_0 + \hat{b}_1 X_1 + \hat{b}_2 X_2 + \varepsilon$, де $\hat{b}_0$, $\hat{b}_1$ та $\hat{b}_2$ є оцінками параметрів. Причому змінна ε повинна бути випадковою величиною з нормальним розподілом. Нульова гіпотеза про нормальність розподілу ε була перевірена за критерієм Пірсона. Висунута гіпотеза була відкинута для рівня значимості 0,05. Це свідчить про те, що розподіл ε не є гаусівським та вказує на відсутність теоретичного обґрунтування застосування лінійної регресійної моделі. Тому слід побудувати нелінійну регресійну модель.

Нелінійна регресійна модель має вигляд $Y = 10^{\varepsilon + \hat{b}_0} X_1^{\hat{b}_1} X_2^{\hat{b}_2}$, де ε – гаусівська випадкова величина, $\varepsilon \sim N(0, \sigma_\varepsilon^2)$, $\hat{b}_0$, $\hat{b}_1$ та $\hat{b}_2$ є оцінками параметрів, які мають відповідно такі значення: 1,0021, 0,8845, 0,1871 та знайдені за допомогою методу найменших квадратів.

Створена нелінійна регресійна модель була перевірена за допомогою R^2 – множинного коефіцієнта детермінації, MMRE – середньої величини відносної похибки та PRED(0,25) – відсотку прогнозованих результатів, для яких величини відносної похибки MMRE менші за 0,25. Допустимі значення мають складати не більше 0,25 для MMRE, та не менше 0,75 для PRED(0,25), а значення R^2 якомога ближче до 1. Для створеної моделі вказані умови виконуються і це свідчить про високу якість отриманої моделі.

Висновки. В роботі було побудовано нелінійну регресійну модель з двома предикторами на основі десяткового логарифму для оцінювання розміру серверних застосунків, що створюються з використанням фреймворку NestJS. Створена модель має добрі показники якості.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Tan, H.B.K., Zhao, Y., & Zhang, H. (2006). *Estimating LOC for information systems from their conceptual data models*, Software Engineering : the 28th International Conference (ICSE '06). Shanghai, China.
- [2] 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. *Науковий журнал «Вчені записки Таврійського національного університету імені В. І. Вернадського. Серія: Технічні науки»*, 31 (70), 124-131. DOI: 10.32838/2663-5941/2020.1- 1/23
- [3] Приходько, С.Б., Приходько, Н.В., Ворона, М.В., & Беловол, 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.

Latanska L.O., Brynzei O.A.

Nonlinear regression model for estimating the size of server applications created using the NestJS framework

Abstract. In the work, a nonlinear regression model was built for estimating the size of server applications created using the NestJS framework.

Keywords: software size estimation; server applications; nonlinear regression model.

ТРЬОХФАКТОРНА НЕЛІНІЙНА РЕГРЕСІЙНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ РОЗМІРУ WEB-ЗАСТОСУНКІВ НА JAVA З ВИКОРИСТАННЯМ SPRING FRAMEWORK

Латанська Л.О., к.ф.-м.н., доцент¹; Борейко А.В.²

^{1,2} Національний університет кораблебудування імені адмірала Макарова

^{1,2} Україна, Миколаїв

¹ llatanskaya@gmail.com; ² sreadbe44@gmail.com

Анотація. Було отримано емпіричні дані з ресурсу GitHub для побудови трьохфакторної нелінійної регресійної моделі для оцінювання розміру web-застосунків на Java з використанням Spring Framework та було виконано їх передобробку. Буде побудовано трьохфакторну нелінійну регресійну модель для прогнозування можливого розміру вказаного програмного забезпечення.

Ключові слова: трьохфакторна нелінійна регресійна модель; веб-застосунки; оцінювання розміру; передобробка даних.

Вступна частина. Оцінювання трудомісткості розробки програмного забезпечення, в тому числі web-застосунків на Java з використанням фреймворку Spring, є однією з найважливіших задач програмної інженерії.

Одним із підходів для оцінювання трудомісткості розробки програмного забезпечення є параметричне оцінювання. При такому оцінюванні використовуються існуючі моделі, як наприклад, СОСОМО II [1]. Одним з основних факторів в цих моделях є розмір програмного забезпечення, а він, як правило, невідомий на початковому етапі розробки.

Для прогнозування розміру використовуються лінійні регресійні моделі, проте їх можна застосовувати тоді, коли залишки або залежна змінна розподілені за нормальним законом, що буває рідко [2]. В такому випадку використовують нелінійні регресійні моделі.

Виходячи з вище викладеного, побудова трьохфакторної нелінійної регресійної моделі для оцінювання розміру web-застосунків на Java з використанням Spring Framework, є актуальною, а розробка програми для її реалізації має практичну цінність.

Метою роботи є підвищення достовірності оцінювання розміру web-застосунків на Java з використанням Spring Framework, за рахунок побудови трьохфакторної нелінійної регресійної моделі.

Основна частина. Для досягнення поставленої мети виконано наступні дії:

- Проаналізовано моделі, які вже побудовані.

З аналізу останніх публікацій можна зробити висновок про те, що якість моделей залежить від мови програмування, фреймворків, середовища розробки та інших факторів. Моделі для оцінювання розміру web-застосунків, які розроблено з використанням Spring Framework та мови Java, не було знайдено. Тому є сенс розробити таку модель. Для цього:

- Зібрано емпіричні дані з ресурсу GitHub.

Обрано 40 проєктів, які розроблені на мові Java з використанням Spring Framework, та зібрано дані, використовуючи програмне забезпечення СК-metrics([ck home](#)), по кількості класів, середній кількості методів на клас, кількості строк коду та глибині дерева наслідування.

- Виконано передобробку даних.

Використовуючи критерій згоди Пірсона, визначено, що дані не є нормально розподіленими. Для нормалізації використано метод десяткового логарифму [3].

Нормалізовані дані перевірено на наявність дублікатів та викидів. Для виявлення викидів використано квадрат відстані Махаланобіса, тестову статистику та критичне значення Фішера. Ті значення з вибірки, для яких тестова статистика перевищує критичне значення Фішера, є викидами. Було знайдено та вилучено 4 викиди [4].

Планується:

- Побудувати трьохфакторну нелінійну регресійну модель для прогнозування розміру web-застосунків на Java з використанням Spring Framework. Параметри моделі буде визначено методом найменших квадратів.

- Виконати порівняння якості отриманої трьохфакторної нелінійної регресійної моделі з існуючими моделями для фреймворків з тими ж факторами.

- Розробити програмне забезпечення для реалізації побудованої моделі.

Висновки. Виконано аналіз існуючих моделей для прогнозування розміру програмного забезпечення, обґрунтовано необхідність побудови трьохфакторної нелінійної регресійної моделі для оцінювання розміру web-застосунків на Java з використанням Spring Framework, зібрано емпіричні дані для побудови моделі та виконано їх передобробку. Планується побудувати трьохфакторну нелінійну регресійну модель для прогнозування розміру web-застосунків на Java з використанням Spring Framework за допомогою нормалізуючого перетворення – десяткового логарифму, виконати оцінювання прогнозних можливостей отриманої моделі та розробити програмне забезпечення для її реалізації.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Боем, Б. (2000). *Економіка розробки програмного забезпечення*. Prentice Hall.
- [2] 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. *Науковий журнал «Вчені записки Таврійського національного університету імені В. І. Вернадського. Серія: Технічні науки»*, 31 (70), 124-131. DOI: 10.32838/2663-5941/2020.1- 1/23
- [3] Приходько, Н.В., & Приходько, С.Б. (2018). Нелінійна регресійна модель для оцінювання розміру програмного забезпечення інформаційних систем на базі VB. *Інформаційні технології та комп'ютерна інженерія*, 3, 37-42.
- [4] Макарова, Л.М., Приходько, Н.В., & Кудін, О.О. (2019). Побудова нелінійної регресійної моделі для оцінювання розміру веб-застосунків, реалізованих мовою Java. *Вісник Херсонського національного технічного університету*, 2 (69), 145-153.

Liudmyla Latanska, Artem Boreyko

Three factored nonlinear regression model for estimating the size of web-applications on Java using Spring framework

Abstract. Empirical data was collected from the GitHub resource to build a threefactored nonlinear regression model for estimating the size of web-applications on Java using Spring Framework and the preprocessing of it was performed. The threefactored nonlinear regression model for forecasting the size of software will be built.

Keywords: threefactored nonlinear regression model, web-applications, estimating the size, preprocessing of data.

УДК 004.412:519.237.5

МАТЕМАТИЧНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ СКЛАДНОСТІ ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОЕКТУВАННЯ ЧЕРЕЗ ЗВ'ЯЗКИ МІЖ КЛАСАМИ КОМП'ЮТЕРНИХ ІГОР

Приходько С.Б., д.т.н., професор¹; Папакін М.М.²

^{1,2} Національний університет кораблебудування імені адмірала Макарова

^{1,2} Україна, Миколаїв

¹sergiy.prykhodko@nuos.edu.ua; ²211263@nuos.edu.ua

***Анотація.** В роботі на основі перетворення у вигляді десяткового логарифму побудовано рівняння еліпсу прогнозування для оцінювання складності об'єктно-орієнтованого проектування через зв'язки між класами комп'ютерних ігор.*

***Ключові слова:** рівняння; еліпс прогнозування; оцінювання складності об'єктно-орієнтованого проектування; зв'язки між класами; комп'ютерна гра.*

Вступна частина. Розробка комп'ютерних ігор є складним завданням, яке вимагає технічної майстерності та навичок інтеграції з кількох областей комп'ютерних наук [1]. Фактично, багато основних традиційних викликів розробки програмного забезпечення (ПЗ) виникають під час розробки комп'ютерних ігор як складних програмних систем, таких як інженерія великомасштабного ПЗ, розробка вимог до комп'ютерних ігор, об'єктно-орієнтоване проектування (ООП) ігрового ПЗ тощо [1]. Однак розробка комп'ютерних ігор – це не те ж саме, що розробка традиційних програмних систем [2], що викликає певні проблеми під час їх виробництва. А це у свою чергу потребує контролю складності ООП в процесі розробки ПЗ, в тому числі оцінювання складності ООП через зв'язки між класами.

Згідно з [3] для оцінювання складності ООП через зв'язки між класами можуть бути використані дві програмні метрики: зв'язаність між об'єктами (CBO, Coupling Between Objects) та відгук класу (RFC, the Response for Class). Але на практиці, як правило, ці метрики аналізуються окремо без врахування кореляції між ними, а прийнятні їх значення визначаються лише на рівні класу. Так для класу, значення CBO між 1 та 5 вважається добрим, а рекомендоване значення RFC складає від 0 до 50. Відповідно до [4], значення CBO більше 14 є занадто високим. Тому виникає потреба у математичній моделі, яка би дозволяла здійснювати оцінювання складності ООП через зв'язки між класами за значеннями CBO та RFC з урахуванням їх кореляції на рівні застосунку в цілому, включаючи і комп'ютерні ігри, що розробляються із використання об'єктно-орієнтованих мов програмування.

Метою дослідження є побудова математичної моделі для оцінювання складності ООП через зв'язки між класами в залежності від метрик RFC та CBO комп'ютерних ігор, що розробляються із використання об'єктно-орієнтованих мов програмування.

Основна частина. Як і в [5], у якості такої математичної моделі для оцінювання складності ООП через зв'язки між класами пропонується використовувати рівняння трансформованого еліпсу передбачення в залежності від метрик RFC та CBO. Як відомо [5], еліпс передбачення використовують для визначення викидів у двовимірних гаусівських даних. Але двовимірний розподіл даних метрик RFC та CBO для комп'ютерних ігор не є гаусівським. Так, ми перевірили двовимірні дані метрик RFC та CBO з 58 комп'ютерних ігор [6] на нормальність за допомогою критерію Мардіа (Mardia's test), якій базується на багатовимірних асиметрії та ексцесі. За цим критерієм двовимірний розподіл зазначених

даних не є гаусівським. Тому, як і в [5], ми використовуємо трансформований еліпс передбачення, який будується наступним чином. Спочатку за допомогою взаємо-зворотного нормалізуючого перетворення двовимірні негаусівські дані нормалізують таким чином щоб розподіл нормалізованих даних був наближений до гаусівського. Далі для нормалізованих даних будують еліпс передбачення. І нарешті, за еліпсом передбачення для нормалізованих даних із використанням зворотного до нормалізуючого перетворення отримують трансформований еліпс передбачення. Вихід точки даних за трансформований еліпс передбачення буде вказувати на наявність викиду з певною довірчою ймовірністю, для якої цей еліпс було побудовано. Також вихід точки даних із значеннями метрик RFC та СВО на рівні застосунку за еліпс буде вказувати і на високу складність через зв'язки між класами для цього застосунку. І навпаки, якщо зазначена точка знаходиться всередині за еліпсу, то вказана складність є невисокою. Для нормалізації двовимірних даних метрик RFC та СВО з 58 комп'ютерних ігор ми використовували перетворення у формі десяткового логарифму. У зазначеному рівнянні еліпсу квадрат відстані Махаланобісу (Mahalanobis) для десяткових логарифмів метрик RFC та СВО на рівні застосунку дорівнює 12,08.

Висновки. У якості математичної моделі для оцінювання складності ООП через зв'язки між класами запропоновано використовувати рівняння трансформованого еліпсу передбачення, яке було побудоване на основі перетворення у формі десяткового логарифму. Для комп'ютерних ігор невисока складність ООП через зв'язки між класами характеризується виконанням наступної умови: квадрат відстані Махаланобісу для десяткових логарифмів метрик RFC та СВО на рівні застосунку менше або дорівнює 12,08.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Scacchi, W., & Cooper, K.M. (2015). Research challenges at the intersection of computer games and software engineering. *Foundations of Digital Games : 2015 Conference (FDG)* (pp. 1-8). Pacific Grove, CA, USA.
- [2] Furtado, A.W., Santos, A.L., Ramalho, G.L., & de Almeida, E.S. (2011). Improving digital game development with software product lines. *IEEE software*, 28(5), 30–37.
- [3] Chidamber, S., & Kemerer, C. (1994). A metrics suite for object oriented design. *IEEE Transactions on Software Engineering*, 20(6), 476-493.
- [4] Sahraoui, H.A., Godin, R., & Miceli, T. (2000). Can metrics help to bridge the gap between the improvement of OO design quality and its automation? *Software Maintenance : 2000 International Conference (ICSM)* (pp. 154-162). San Jose, CA, USA. DOI: <https://doi.org/10.1109/ICSM.2000.883034>
- [5] Приходько, С.Б., Смикодуб, Т.Г. (2021). Математична модель для перевірки взаємозв'язків між частинами Java-застосунків за метриками RFC та СВО. *Інформаційні технології: моделі, алгоритми, системи : II-га Всеукраїнська науково-практична інтернет конференція (ITMAS – 2021)* (С. 25-26). Миколаїв, Україна. <http://itconf.nuos.edu.ua/2021/proceedings/>
- [6] Sabhat, N., Malik, A.A., & Azam, F. (2017). Utility of CK Metrics in Predicting Size of Board-Based Software Games. *Mehran University Research Journal of Engineering and Technology*, 36(4), 975-986. DOI: <https://doi.org/10.22581/muet1982.1704.22>

Prykhodko S.B, Papakin M.M.

A mathematical model for evaluating the complexity of object-oriented design due to the relationships between classes of computer games

Abstract. In the work, based on the transformation in the form of a decimal logarithm, a prediction ellipse is built for evaluating the complexity of object-oriented design due to the relationships between classes of computer games.

Keywords: equation; prediction ellipse; evaluating the object-oriented design complexity; relationships between classes; computer game.

УДК 004.412:519.237.5

АНАЛІТИЧНА МОДЕЛЬ ЧОТИРИВИМІРНОГО ЗАКОНУ РОЗПОДІЛУ МЕТРИК ВЕБ ЗАСТОСУНКІВ, ЩО СТВОРЮЮТЬСЯ ЗА ДОПОМОГОЮ ФРЕЙМВОРКУ CAKEPHP

Приходько А.С.¹¹ Одеський національний університет імені І. І. Мечникова¹ Україна, Одеса¹whiterandrek@gmail.com

Анотація. В роботі на основі багатовимірного розподілу Джонсона для сімейства S_B було побудовано аналітичну модель чотиривимірного закону розподілу метрик веб застосунків, що створюються з використанням фреймворку CakePHP.

Ключові слова: аналітична модель; чотиривимірний розподіл; програмна метрика; веб застосунок; фреймворк.

Вступна частина. Як відомо, закон розподілу є універсальною характеристикою випадкової величини або сукупності випадкових величин. Підбор аналітичної моделі закону розподілу є засобом узагальненого представлення емпіричних даних у тих випадках, коли відсутнє теоретичне обґрунтування закону розподілу випадкових величин [1]. Інформація про багатовимірний закон розподілу в першу чергу потрібна для визначення наявності викидів у багатовимірних наборах даних, а також для побудови математичних моделей для оцінювання певних програмних метрик, зокрема кількості строк коду веб застосунків [2-5]. Зараз багато веб застосунків створюються за допомогою різних фреймворків, які прискорюють розробку ПЗ, одним з яких є CakePHP. Це потребує побудови відповідних аналітичних моделей закону розподілу для метрик програмних застосунків, у тому числі і веб застосунків, що створюються з використанням фреймворку CakePHP.

Метою дослідження є побудова аналітичної моделі чотиривимірного закону розподілу для метрик веб застосунків, що створюються з використанням фреймворку CakePHP.

Основна частина. На теперішній час для опису емпіричних даних аналітичну модель закону розподілу підбирають зазвичай або із сімей розподілів Джонсона, або із класів розподілів Пірсона. В [1] рекомендується застосовувати розподіли Джонсона. Їх перевага полягає у тому, що вони приводять після певних перетворень до нормально розподіленої випадкової величини. А історично склалося так, що велика кількість статистичних критеріїв та методів оцінювання розроблені в основному тільки для випадку нормального початкового розподілу.

Розглянемо побудову аналітичної моделі чотиривимірного закону розподілу для метрик веб застосунків, що створюються з використанням фреймворку CakePHP. Враховуючи на те, що CakePHP є PHP фреймворком, метрики 38 веб застосунків, які розміщені на сайті GitHub (<https://github.com>), були отримані за допомогою інструменту PhpMetrics (<https://phpmetrics.org/>). Ми брали такі метрики: розмір веб-застосунку Y у тисячах рядків коду; кількість класів X_1 ; середню кількість методів на клас X_2 ; глибину дерева спадкування X_3 . Ці метрики були обрані, тому що їх можна отримати з діаграми класів та між ними немає мультиколінеарності. Далі ми перевірили багатовимірні дані на нормальність за допомогою тесту Мардіа (Mardia's test), якій 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. Нормалізацію даних ми здійснювали за допомогою чотиривимірного перетворення Джонсона сімейства S_B , оцінювання параметрів якого здійснювалося за методом максимальної правдоподібності. Використання цього методу потребує інформації про логарифмічну функцію правдоподібності, побудова якої здійснюється на основі відповідного закону розподілу у вигляді щільності ймовірності. На основі багатовимірного розподілу Джонсона для сімейства S_B було побудовано чотиривимірну щільність ймовірності для зазначених вище метрик веб застосунків, що створюються з використанням фреймворку CakePHP. Це дало змогу визначити наявність 8 викидів у даних.

Висновки. В роботі на основі багатовимірного розподілу Джонсона для сімейства S_B було побудовано аналітичну модель чотиривимірного закону розподілу у вигляді чотиривимірної щільності ймовірності для метрик веб застосунків, що створюються з використанням фреймворку CakePHP. Ця модель в першу чергу може бути застосована для визначення наявності викидів у чотиривимірних даних з метрик веб застосунків, що створюються з використанням фреймворку CakePHP. В подальшому планується побудова аналітичних моделей чотиривимірного закону розподілу у вигляді чотиривимірної щільності ймовірності для метрик веб застосунків, що створюються з використанням інших PHP фреймворків.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Коваленко, І.І., Приходько, С.Б., & Латанська, Л.О. (2011). *Сучасні методи статистичного аналізу даних*. Миколаїв, НУК.
- [2] Приходько, С.Б., Приходько, Н.В., Ворона, М.В., & Беловол, І.О. (2021). Нелінійна регресійна модель для оцінювання розміру Web-застосунків, що створюються з використанням фреймворку Laravel. *Інформаційні технології та комп'ютерна інженерія*, 1 (50), 115-121. DOI: <https://doi.org/10.31649/1999-9941-2021-50-1-115-121>
- [3] Prykhodko, S.B., Shutko, I.S., & Prykhodko, A.S. (2021). A nonlinear regression model to estimate the size of Web apps created using the CakePHP framework. *Radio Electronics, Computer Science, Control*, 4 (59). 129-139. DOI: <http://dx.doi.org/10.15588/1607-3274-2021-4-12>
- [4] Prykhodko, S., Shutko, I., & Prykhodko, A. (2021). Early LOC estimation of Web apps created using Yii framework by nonlinear regression models. *WSEAS Transactions on Computers*, 20, 321-328. DOI: <http://dx.doi.org/10.37394/23205.2021.20.35>
- [5] Prykhodko, S.B., Shutko, I.S., & Prykhodko, A.S. (2022). Early size estimation of web apps created using Codeigniter framework by nonlinear regression models. *Radioelectronic and computer systems*, 3 (103). 84-94. DOI: <https://doi.org/10.32620/reks.2022.3.06>

Prykhodko A.S.

Analytical model of the four-dimensional distribution law of metrics of web applications created using the CakePHP framework

Abstract. In the work, an analytical model of the four-dimensional distribution law of web application metrics created using the CakePHP framework was built based on Johnson's multidimensional distribution for the S_B family.

Keywords: analytical model; four-dimensional distribution; software metrics; web application; framework.



УПРАВЛІННЯ ПРОЕКТАМИ І ПРОГРАМАМИ

ПРОГРАМНІ ЗАСОБИ МОНІТОРИНГУ ПРОГРЕСУ ПРОЄКТУ В УМОВАХ НЕПОВНОЇ ІНФОРМАЦІЇ

Вербицький В.О.¹; Гусєва Ю.Ю., д.т.н., доцент²

^{1,2} Харківський національний університет міського господарства імені О.М. Бекетова

^{1,2} Україна, Харків

¹ vladyslav.verbytskyi@kname.edu.ua ; ² yulia.guseva@kname.edu.ua

Анотація. Під час дослідження розроблено програмний інструмент моніторингу прогресу проєкту на основі використання теорії сірих систем, що дає змогу здійснювати відстеження виконання проєкту в умовах неповної інформації. Продемонстровано результати використання розробки на прикладі умовного проєкту.

Ключові слова: метод освоєного обсягу; проєкт; моніторинг; теорія сірих систем.

Вступна частина. Використання методів та інструментів моніторингу прогресу проєкту є запорукою можливості своєчасного виявлення відхилень та адекватної реакції на них. Традиційним методом відстеження виконання проєкту є Earned Value Method (EVM) [1], який дозволяє відстежувати виконання планів з розкладу і вартості проєкту у грошовій формі. Але проблемою є використання цього методу в умовах невизначеності та неповної інформації, що є досить типовою ситуацією для проєктного управління. Відомою спробою вирішення цієї проблеми є використання теорії нечітких множин, але це викликає необхідність побудови функцій приналежності для всіх задач проєкту. Ще одним шляхом нівелювання проблеми неповноти інформації є використання теорії сірих систем (grey theory) [2, 3].

Метою дослідження є розробка інструментарію модифікованого за рахунок використання теорії сірих систем EVM.

Основна частина. В роботі [2] наведено формули для розрахунку «сірого» освоєного обсягу $\otimes EV$ та «сірих» індексів $\otimes CPI$ (виконання проєкту за вартістю) та $\otimes SPI$ (виконання проєкту за розкладом):

$$\otimes EV_i = \otimes AP_i \cdot BAC_i = [BAC_i \cdot \underline{AP_i}, BAC_i \cdot \overline{AP_i}], \quad (1)$$

де $\otimes AP_i$ – відсоток повноти виконання і-ї роботи;

BAC_i – бюджет по закінченню і-ї роботи.

$$\otimes EV = \sum_{i=1}^n \otimes EV_i = \left[\sum_{i=1}^n \underline{EV_i}, \sum_{i=1}^n \overline{EV_i} \right], \quad (2)$$

$$\otimes SPI = \frac{\otimes EV}{PV} = \left[\frac{\underline{EV}}{PV}, \frac{\overline{EV}}{PV} \right], \quad (3)$$

де PV – планова вартість і-ї роботи;

$$\otimes CPI = \frac{\otimes EV}{AC} = \left[\frac{\underline{EV}}{AC}, \frac{\overline{EV}}{AC} \right], \quad (4)$$

де AC – фактична вартість і-ї роботи;

Для програмної реалізації методу в середовищі MS Project було створено дві нові розрахункові таблиці – Grey SPI та Grey CPI (рис. 1 та рис. 2).

Приклад розрахунку $\underline{SPI_i}$ представлено на рисунку 3.

Результати використання розроблених таблиць показано для умовного проєкту впровадження системи процесного управління в організації. Діаграму Ганта з основними етапами проєкту представлено на рисунку 4.

Название	БСЗР	AC	EV-	EV+	SPI-	SPI+	Авг	Сен	Окт
Впровадження BPM	26 475,00 ₴	0	0	0	0	0			
Початок проєкту (прийняття рішення)	0,00 ₴	0	0	0	1	1			
Проведення діагностики СУ	5 000,00 ₴	4500	5000	5000	1	1			
Формування мережі процесів та призначення власників	4 200,00 ₴	0	0	0	0	0			
Детальне планування і організація робіт	11 900,00 ₴	0	0	0	0	0			
Детальне планування і організація роботи	1 500,00 ₴	1350	1350	1500	0,9	1			
Створення робочих груп за кожним процесом	2 000,00 ₴	1000	1800	1900	0,9	0,95			
Навчання робочих груп методам процесного підходу	6 000,00 ₴	4000	3600	4200	0,6	0,7			
Планування діяльності робочих груп	2 400,00 ₴	2400	0	0	0	0			

Рисунок 1 – Фрагмент таблиці Grey SPI

Название	БСЗР	AP-	AP+	EV-	EV+	SPI-	SPI+	Авг	Сен	Окт
Впровадження BPM	26 475,00 ₴	0	0	0	0	0	0			
Початок проєкту (прийняття рішення)	0,00 ₴	1	1	0	0	1	1			
Проведення діагностики СУ	5 000,00 ₴	1	1	5000	5000	1	1			
Формування мережі процесів та призначення власників	4 200,00 ₴	0	0	0	0	0	0			
Детальне планування і організація робіт	11 900,00 ₴	0	0	0	0	0	0			
Детальне планування і організація роботи	1 500,00 ₴	0,9	1	1350	1500	0,9	1			
Створення робочих груп за кожним процесом	2 000,00 ₴	0,9	0,95	1800	1900	0,9	0,95			
Навчання робочих груп методам процесного підходу	6 000,00 ₴	0,6	0,7	3600	4200	0,6	0,7			

Рисунок 2 – Фрагмент таблиці Grey CPI

Формула для 'SPI-'

Изменение формулы

SPI- =

$$\text{IIF}([БСЗР]=0;1;[EV-]/[БСЗР])$$

Вставка: Поле Функция Импорт формулы...

Справка OK Отмена

Рисунок 3 – Приклад розрахунку значень таблиці Grey SPI

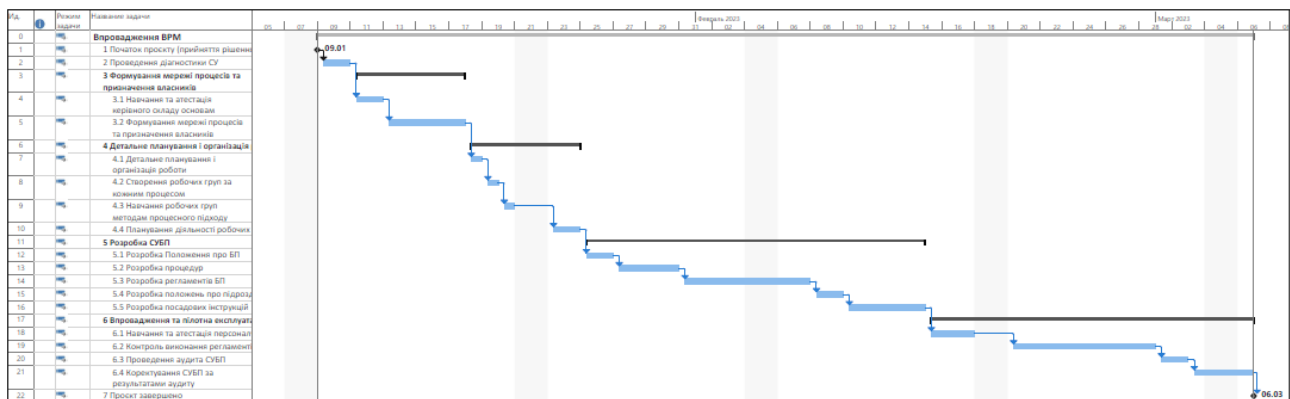


Рисунок 4 – Діаграма Ганта проєкту впровадження системи процесного управління

Відзначимо, що однією з головних переваг використання методу «сірого» EVM є те, що він враховує невизначений характер вхідних даних і використовує досить прості обчислення. Іншою значною перевагою використання теорії сірих систем для EVM є зрозумілість вихідних даних для всіх стейкхолдерів проєкту.

Висновки. Таким чином, розроблено програмний інструмент моніторингу прогресу проєкту на основі використання теорії сірих систем, що дає змогу здійснювати відстеження

виконання проєкту в умовах неповної інформації. Продемонстровано результати використання розробки на прикладі умовного проєкту.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] The Practice Standard for Earned Value Management – Second Edition. (2019). Project Management Institute.
- [2] Mahmoudi, A., Bagherpour, M. & Javed, S.A. (2021). Grey Earned Value Management: Theory and Applications. *IEEE Transactions on Engineering Management*, 6 (68), 1703-1721.
- [3] Гусєва, Ю.Ю., Вербицький, В.О., Чумаченко, І.В. (2022). Інструменти теорії сірих систем для контролю виконання вимог стейкхолдерів та досягнення цінності проєкту. *Міжнародна науково-практична конференція «Інтелектуальні інформаційні системи в управлінні проєктами та економіці в умовах воєнного стану», Коблево, 13-16 вересня 2022 р. Праці. Харків. ХНУРЕ.*

Verbytskyi V.O., Husieva Yu.Yu.

Project progress monitoring software in the conditions of incomplete information

Abstract. During the research, a software tool for project progress monitoring was developed based on the use of grey theory, which enables tracking of project implementation in conditions of incomplete information. The results are demonstrated on the example of a conditional project.

Keywords: Earned Value Method; project; monitoring; grey theory.

НАПРЯМКИ РОЗВИТКУ УПРАВЛІННЯ ПРОЄКТАМИ МАРКЕТИНГ-ЛОГІСТИЧНОГО ЗАБЕЗПЕЧЕННЯ ЛАНЦЮГІВ ПОСТАЧАННЯ ФАРМАЦЕВТИЧНИХ ТОВАРІВ

Гусєва-Божаткіна В. А.¹; Єжгуров Р.В.²

^{1,2} Національний університет кораблебудування імені адмірала Макарова

^{1,2} Україна, Миколаїв

¹gusevabozh@meta.ua; ¹ ORCID: 0000-0002-1117-3391

²11123111leo@gmail.com

Анотація. В даній роботі сформульовані основні проблеми функціонування ланцюгів постачань фармацевтичної продукції. Досліджено розвиток логістичного управління за допомогою використання інноваційних цифрових технологій. Визначена роль стратегічного управління в ефективному реагуванні на системні збої у логістичному забезпеченні ланцюгів постачань фармацевтичних товарів.

Ключові слова: ланцюги постачань; логістичний менеджмент; цифрові технології; інтернет речей; стратегічне управління.

Вступна частина. В основі ефективного задоволення потреб споживача лежить надання якісного товару за оптимальною ціною, на що, значною мірою, впливає організація управління проєктами маркетингових та логістичних процесів продовж усього ланцюга постачань на шляху просування товару від виробника до споживача.

Особливо це стосується управління проєктами постачань фармацевтичних товарів, коли зростають вимоги і потреби клієнтів та зростає конкуренція з боку ринкових суперників, це змушує сучасні фармацевтичні установи до пошуку оптимальних рішень в рамках ланцюгів постачань. Такі рішення призводять до посилення розподілу функцій між окремими ланками ланцюга, зменшення операційних витрат, підвищення якості інформації, пов'язаної зі здійснюваними діями, інтеграції дій з метою більш досконалого обслуговування кінцевого клієнта і здобуття конкурентної переваги, чому сприяє створення умов забезпечення функціонування ланцюгів постачань товарів споживчого попиту маркетинговим і логістичним інструментарієм.

Управління проєктами маркетинг-логістичного забезпечення ланцюгів постачання є відносно новим напрямком. Формування сучасних ланцюгів постачання фармацевтичних товарів потребує формування розвиненої системної методології управління такими ланцюгами, а її розробка має спиратися на відповідне маркетинг-логістичне забезпечення. Ланцюг постачань (supply chain) представляє безліч ланок, пов'язаних між собою інформаційними, грошовими та товарними потоками. Ланцюжок постачання лікарських засобів споживачу (пацієнту) включає не менше трьох учасників фармацевтичного ринку (рисунок 1).

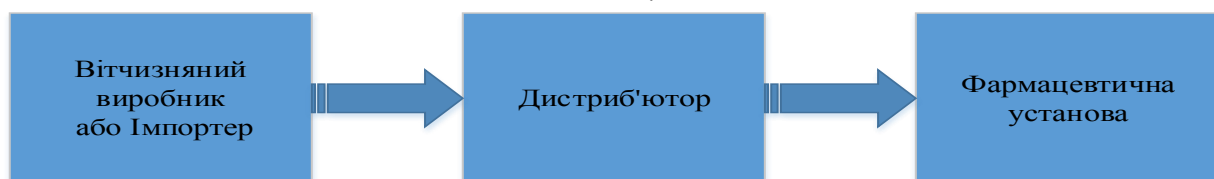


Рисунок 1 – Ланцюжок постачання лікарських засобів

Метою дослідження є розгляд напрямів розвитку управління проєктами маркетинг-логістичного забезпечення ланцюгів постачання фармацевтичних товарів, виходячи із їх характеристик, сутності та особливості формування ланцюга постачання в умовах можливостей «е-ланцюга постачання», та орієнтованістю на соціальну спрямованість ліків.

Основна частина. Система охорони здоров'я в сучасних умовах залишається одним із пріоритетних напрямків діяльності суспільства і держави. Потужну нішу в системі охорони здоров'я займає фармацевтична сфера. Поширення пандемії COVID-19 та військових дій, призвели до значних збурень у світовій торгівлі на цільових ринках фармпродукції, ускладнили кон'юнктуру українського ринку фармпродукції на внутрішніх та зовнішніх ринках, логістику експортно-імпортних операцій, позначилися на результативності та ефективності функціонування ланцюгів постачання фармпродукції, загострили конкурентну ситуацію на цільових ринках фармтоварів. Слід зазначити, що пандемія та військові дії суттєво змінили структуру ринку фармпродукції в частині збільшення частки госпітального сегмента.

Для поліпшення результативності функціонування ланцюгів постачань фармпродукції проаналізуємо зміни в логістичній сфері, спричинені пандемією COVID-19 та військовими діями.

Реагування на системні збої у функціонуванні логістичних ланцюгів постачань фармпродукції передбачає послідовне виконання дій за алгоритмом наведеному на рисунку 2.

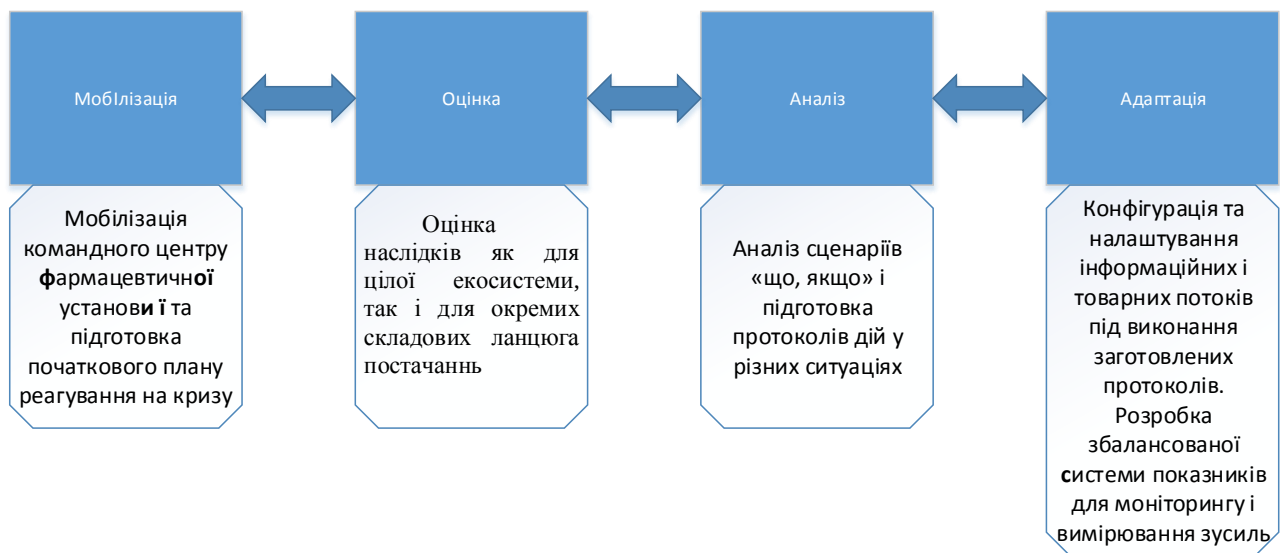


Рисунок 2 – Алгоритм реагування менеджменту фармацевтичної установи на кризові ситуації у функціонуванні ланцюгів постачань

Оцінюючи перспективи виходу з коронавірусної кризи, варто зазначити, що майбутнє за цифровими технологіями. Для посилення ланцюгів постачань можливо використовувати наступні інноваційні цифрові інструменти: електронна комерція; відстеження в режимі реального часу; подолання логістичних перешкод за допомогою технологій штучного інтелекту (AI); безпілотні транспортні засоби і дрони; цифрова мережа постачання; мобільні технології і додатки та ін. [1]. Управління проєктами в ланцюгах постачання фармпродукції являє собою застосування концепції і методів управління проєктами в процесі створення / перетворення ланцюгів постачання, а також вирішення завдань по їх функціонуванню. Проєктування ланцюгів постачань здійснюється за відповідними критеріями. Найбільш часто таким критерієм є логістичні витрати. Розробка проєктів з різними варіантами економії витрат в логістиці – це актуальний напрямок розвитку управління ланцюгами постачань фармпродукції з використанням програмного забезпечення в умовах диджиталізації [2].

Як відомо з [3] транспортування ліків з фармацевтичних заводів до аптек — це індустрія обсягом 78,8 мільярда доларів, згідно з Pharmaceutical Commerce's annual Biopharma Cold Chain Sourcebook. Традиційно лише витрати на “холодний ланцюг” — спостереження за температурою та вологістю — становлять 12,6 мільярда доларів.

Сучасні досягнення технології інтернет речей (IoT) для відстеження, комунікацій та управління ланцюжками постачань ліків допомагають отримати більш ефективний фармацевтичний логістичний менеджмент завдяки комплексним рішенням. Це вигідно як фармацевтичній промисловості, так і споживачам. Технологічні досягнення інтернету речей забезпечують фармацевтичний логістичний менеджмент новими рішеннями, які легко приймати та які стосуються багатьох проблем, пов'язаних з перевезенням ліків. Широке розповсюдження мобільних пристроїв впливає на масштаб впровадження IoT. Інтуїтивно зрозумілі мобільні додатки розширюють можливості логістики в ефективному менеджменті ланцюгів постачань, надаючи доступ до значущих даних.

Висновки. Розв'язання проблем у сфері маркетинг-логістичного забезпечення ланцюгів постачань фармацевтичних товарів, зумовлених пандемією COVID-19 та військовими діями, можливо при ефективному стратегічному управлінні та використанні інноваційних цифрових технологій. Цифрові технології забезпечують прозорість як усього ланцюга постачань фармацевтичної продукції загалом, так і окремих його елементів, дозволяють ефективніше визначати зони потенційного ризику, швидше реагувати на збої.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Пасічник, Т.О. (2021). Міжнародні ланцюги поставок в умовах ковідної кризи. *Економіка та суспільство*. Випуск # 33. DOI: <https://doi.org/10.32782/2524-0072/2021-33-38>
- [2] Нініч, В.З., Швець, А.В., Гриценко, С.І. (2022). Програмні засоби управління проєктами в ланцюгах постачання шляхом диджиталізації. *Збірник доповідей XIX Міжнародної науково-практичної конференції «Проблеми підготовки професійних кадрів з логістики в умовах глобального конкурентного середовища», Національний авіаційний університет, Київ, Україна.* (с. 103-107).
- [3] Інтернет речей змінює фармацевтичний логістичний менеджмент. (2017, 24 травня). <http://tinker.uamper.com/news/nternet-rechey-zmnyu-farmacevtichniy-logstichniy-menedzhment.html>

Guseva-Bozhatkina, V.A., Yezhhurov, R.V.

Directions of development of marketing-logistics project management of pharmaceutical supply chains

Abstract. In this work, the main problems of the functioning of pharmaceutical product supply chains are formulated. The development of logistics management using innovative digital technologies is studied. The role of strategic management in effective response to system failures in the logistical support of supply chains of pharmaceutical goods is defined.

Keywords: supply chains; logistics management; digital technologies; Internet of Things; strategic management.



ІНФОРМАЦІЙНІ СИСТЕМИ ТА ТЕХНОЛОГІЇ

УДК 004.12

ДОСЛІДЖЕННЯ ПРОЦЕСІВ ІНФОРМАЦІЙНОЇ ПІДТРИМКИ ОСВІТНЬОЇ ДІЯЛЬНОСТІ

Маршак О.І.¹; Бобровський О.С.²; Коваль С.С., к.ф.-м.н., доц.³; Ворона М.В., PhD⁴

^{1,2,3,4} Національний університет кораблебудування імені адмірала Макарова

^{1,2,3,4} Україна, Миколаїв

¹olena.marshak@nuos.edu.ua; ²oleksii.bobrovskyi@nuos.edu.ua;

³sergiy.koval@nuos.edu.ua; ⁴mykhailo.vorona@nuos.edu.ua

Анотація. Проаналізовано шляхи підвищення ефективності управління та організації інформаційної підтримки взаємодії стейкхолдерів освітньої діяльності. Процес розглянуто на прикладі розробки веб-сервісів для залучення зацікавлених сторін до визначення цілей освітніх програм та формулювання програмних результатів навчання.

Ключові слова: інформаційна підтримка; веб-сервіс; освітня діяльність; он-лайн обговорення; стейкхолдери; освітня програма.

Вступна частина. Реформування вищої освіти і наукової сфери передбачатиме вдосконалення науково-технічної бази та інформаційного забезпечення, комерціалізацію діяльності, створення системи стимулювання інноваційної діяльності, забезпечення зміни поколінь у науці, зміцнення кадрового потенціалу. Реалізація вищезазначених напрямів можлива за умови інформаційної підтримки освітнього процесу на усіх структурних рівнях закладів вищої освіти, зокрема кафедрах, які забезпечують підготовку здобувачів освіти за освітніми програмами.

Кафедра інформаційних управляючих систем і технологій (ІУСТ) Національного університету кораблебудування імені адмірала Макарова є провідною кафедрою з підготовки здобувачів вищої освіти в галузі ІТ.

Мета дослідження – на прикладі кафедри інформаційних управляючих систем та технологій (ІУСТ) Національного університету кораблебудування імені адмірала Макарова розглянути основні принципи інформаційної підтримки освітньої діяльності з розробкою загальної структури веб-сервісу для он-лайн обговорення.

Основна частина. Методологічну базу досліджень становлять основні наукові положення цифрової трансформації освіти [1], відповідність освітньої діяльності Критеріям оцінювання якості освіти [2], сучасний досвід застосування інформаційних технологій для організації ефективної інформаційної підтримки освітньої діяльності [3]. Нормативною підставою є законодавча база України, а саме Закони України: «Про освіту», «Про вищу освіту» та «Про внесення змін до деяких законів України щодо функціонування інтегрованих інформаційних систем у сфері освіти».

Одним із основних завдань дослідження є аналіз предметної галузі – освітньої діяльності кафедри ІУСТ Національного університету кораблебудування імені адмірала Макарова та її організаційного супроводу. Важливою складовою інформаційної підтримки освітньої діяльності кафедри є інтеграція взаємодії усіх стейкхолдерів освітнього процесу за усіма освітніми програмами кафедри (таблиця 1), де здійснюється підготовка здобувачів освіти за 6 освітніми програмами трьох спеціальностей.

Таблиця 1

Освітні програми кафедри ІУСТ
Національного університету кораблебудування імені адмірала Макарова

Спеціальність	Назва освітньої програми	
	Ступінь вищої освіти магістр	Ступінь вищої освіти бакалавр
122 Комп'ютерні науки	Комп'ютерні науки (КН Б)	Комп'ютерні науки (КН М)
124 Системний аналіз	Системний аналіз (СА Б)	Системний аналіз (СА М)
126 Інформаційні системи та технології	Інформаційні системи та технології (ІСТ Б)	Інформаційні системи та технології (ІСТ М)

До кола стейкхолдерів однієї освітньої програми входять: здобувачі освіти, випускники, адміністрація ЗВО, академічна спільнота, потенційні роботодавці. Найефективнішим способом організації освітньої діяльності кафедри є залучення всіх зацікавлених сторін до визначення цілей освітньої програми та формулювання програмних результатів навчання [2] засобами сайту кафедри. Для оптимізації процесів інформаційної підтримки освітньої діяльності запропоновано розробку веб-сервісів он-лайн обговорення освітніх програм кафедри із подальшою інтеграцією розроблених сервісів до сайту кафедри ІУСТ Національного університету кораблебудування імені адмірала Макарова. Фундаментальним етапом проєктування є розробка загальної структури (рисунок 1) веб-сервісу для он-лайн обговорення освітньої діяльності кафедри, яка потім кастомізується для кожного з веб-сервісів.

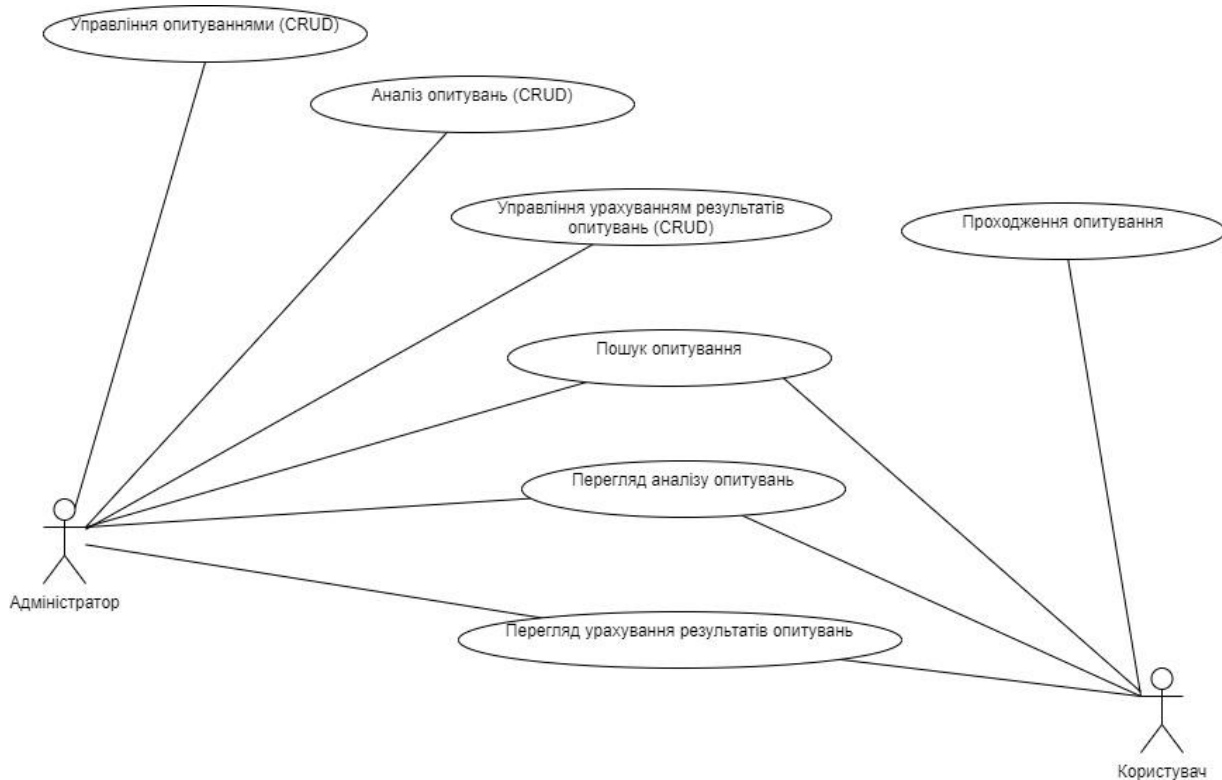


Рисунок 1. Діаграма варіантів використання [складено авторами]

Адміністратор здійснює управління опитуваннями, аналіз опитувань та управління урахуванням результатів опитувань за чотирма основними операціями постійного зберігання CRUD – Create (створення), Read (читання), Update (оновлення), Delete (видалення). Користувачеві надаються можливості пошуку і проходження опитувань, перегляду аналізу та урахування результатів опитувань.

Перспективи подальших досліджень пов'язані з апробацією розроблених веб-сервісів та запуском пілотного проєкту он-лайн залучення всіх зацікавлених сторін до визначення цілей освітньої програми та формулювання програмних результатів навчання.

Висновки.

1. Розроблено структуру веб-сервісу он-лайн взаємодії стейкхолдерів освітньої діяльності.

2. На прикладі веб-сервісів он-лайн обговорення освітніх програм запропоновано принципи створення інформаційної підтримки освітньої діяльності кафедри ІУСТ Національного університету кораблебудування імені адмірала Макарова.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Navitas Ventures (2017). *Digital Transformation in Higher Education*. https://www.navitasventures.com/wp-content/uploads/2017/08/HE-Digital-Transformation-_Navitas_Ventures_-EN.pdf
- [2] Бутенко, А.П., Єременко, О.В., Квіт, С.М. (Ред.) (2020). Рекомендації щодо застосування критерії оцінювання якості освітньої програми. Київ: ТОВ «Український освітянський видавничий центр «Оріон». 66 с. <https://naqa.gov.ua/wp-content/uploads/2020/12/Рекомендації-щодо-застосування-критеріїв-оцінювання-якості-ОП.pdf>
- [3] Триус, Ю.В., Заспа, Г.О., Кожем'якін, О.С., Аширова, А.В. (2020). Інформаційно-аналітична система підтримки освітньої діяльності структурних підрозділів закладів вищої освіти. *Вісник Черкаського державного технологічного університету*, (4), с. 27–38. DOI: <https://doi.org/10.24025/2306-4412.4.2020.219482>
- [4] Трушляков, Є.І., Слободян, С.О., Лабарткава, А.В., Івата, В.В., Савочкіна, В.В. (Ред.) (2020). Положення про організацію освітнього процесу у Національному університеті кораблебудування імені адмірала Макарова. Миколаїв: НУК. 129 с. <https://nuos.edu.ua/wp-content/uploads/2020/10/Polozhennya-pro-organizaciju-navchalnogo-procesu-NUK-2020-zi-zminami.pdf>.

Marshak O.I; Bobrovskyi O.S; Koval S.S; Vorona M.V.

Research of Processes of Information Support of Educational Activities

Abstract. Ways to increase the efficiency of management and organization of information support for the interaction of stakeholders in educational activities are analyzed. The process is considered on the example of the development of web services for the involvement of interested parties in determining the goals of educational programs and formulating program learning outcomes.

Keywords: information support; web service; educational activity; on-line discussion; stakeholders; educational program.

УДК 004.4:613.6

ПРОЄКТ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ОЦІНКИ ВПЛИВУ УЛЬТРАФІОЛЕТОВОГО ВИПРОМІНЮВАННЯ НА ЕКІПАЖ СУДНА

Фаріонова Тетяна Анатоліївна¹, к.т.н., доцент; Орехов Олександр Станіславович²,
аспірант

Національний університет кораблебудування імені адмірала Макарова

Україна, Миколаїв

¹ ORCID 0000-0003-3384-4712; ² ORCID 0000-0002-0001-0140

Анотація. На основі аналізу впливу ультрафіолетового випромінювання на здоров'я екіпажу судна, існуючих заходів попередження ризику ураження шкіри людини, обґрунтовано необхідність розробки програмного забезпечення у вигляді мобільного застосунку. Представлено функціонал та розроблено алгоритм поведінки програмної системи. Використання мобільного застосунку дозволить спрогнозувати ймовірний вплив уф-випромінювання на екіпаж судна в умовах обмеженості або відсутності зв'язку

Ключові слова: мобільний застосунок; оцінка впливу ультрафіолетового випромінювання; екіпаж судна; індекс ультрафіолетового випромінювання.

Вступна частина Ультрафіолетове (уф) випромінювання – це форма електромагнітного випромінювання, з довжиною хвилі від 10 до 400 нм. При надлишковому уф-випромінюванні з'являється вірогідність появи таких негативних ефектів як сонячні опіки, хвороби очей, послаблення імунної системи, передчасне старіння та ризик раку шкіри.[1, 2]. Члени екіпажів суден піддаються значному впливу сонячної радіації, отримують більше еритемних доз, що в свою чергу, ставить цю професію в групу ризику щодо появи важких наслідків впливу уф-випромінювання. Слід зазначити, що уф-випромінювання має 2 варіанту негативних наслідків: короткострокові – сонячні опіки та теплові удари та довгострокові – пошкодження зору чи рак шкіри, які можуть проявити себе через довгий термін, наприклад через 20 або 40 або більше років [3 – 6]. Враховуючи специфіку роботи екіпажів суден, є необхідність інформування моряків, щодо ризику ушкодження шкіри протягом рейсу. Для визначення рівня уф-випромінювання може бути використано спеціальна апаратура, яка вимагатиме певну кваліфікацію та обслуговування, або визначенням одним із емпіричних способів. Рухаючись шляхом найменших витрат, найбільш простим та швидким способом впровадження інформування щодо загрози сонячних опіків та наслідків тривалого перебування на сонці, є розробка спеціального програмного забезпечення, а саме мобільного застосунку, адже кожна людина має мобільний телефон.

Метою роботи є розробка мобільного застосунку для оцінки впливу ультрафіолетового випромінювання на стан здоров'я екіпажу суден, що дозволить запобігати впливу небезпечних факторів ультрафіолетового випромінювання за рахунок своєчасного інформування членів екіпажу про загрозові умови в залежності від міста знаходження та кліматичних умов, отримати персоналізовану інформацію, на основі типу шкіри за її пігментацією.

Основна частина. Для прогнозування індексу уф-випромінювання, можна використати формулу (1) прогнозування на основі емпіричних даних для чистого неба, виведену Майклом ван Вілі, Полом Фортоном та Хені Келдер. [7, 8], яка додатково враховує вплив коефіцієнту

хмарності, альbedo корабля та океану. Параметр уф-випромінювання матиме наступний вигляд:

$$UVI = f(lat, lng, a, Oz) \cdot f(c) \cdot f_{alb}(ao, as), \quad (1)$$

де, lat - широта, lng - довгота, a - кут положення сонця, Oz - рівень озону, $f(c)$ - функція залежності від коефіцієнту хмарності та $f_{alb}(ao, as)$ функція залежності від альbedo океану (ao) та корабля (as).

Концентрація озону має зворотно пропорційний вплив на рівень уф-випромінювання. Чим менше рівень концентрації озону в атмосфері, тим більше уф хвиль проникає до поверхні землі і тим більший рівень уф-випромінювання. [9]. Кожен тип шкіри можна охарактеризувати параметром мінімальної еритемної дози випромінювання. Цей параметр описує кількість енергії уф-випромінювання, при якому вірогідна поява почервоніння шкіри (еритема). Посилаючись на отримані дані параметру еритемної дози із статі “Обґрунтованість і практична реакція шкіри на Сонце для типів шкіри від I до VI” [9], можна розрахувати приблизний час згорання на сонці для різних типів шкіри.

На основі специфіки предметної галузі та функціональних вимог до програмного забезпечення, авторами запропоновано варіанти використання програмного забезпечення у вигляді Use Case діаграми (рис.1а).

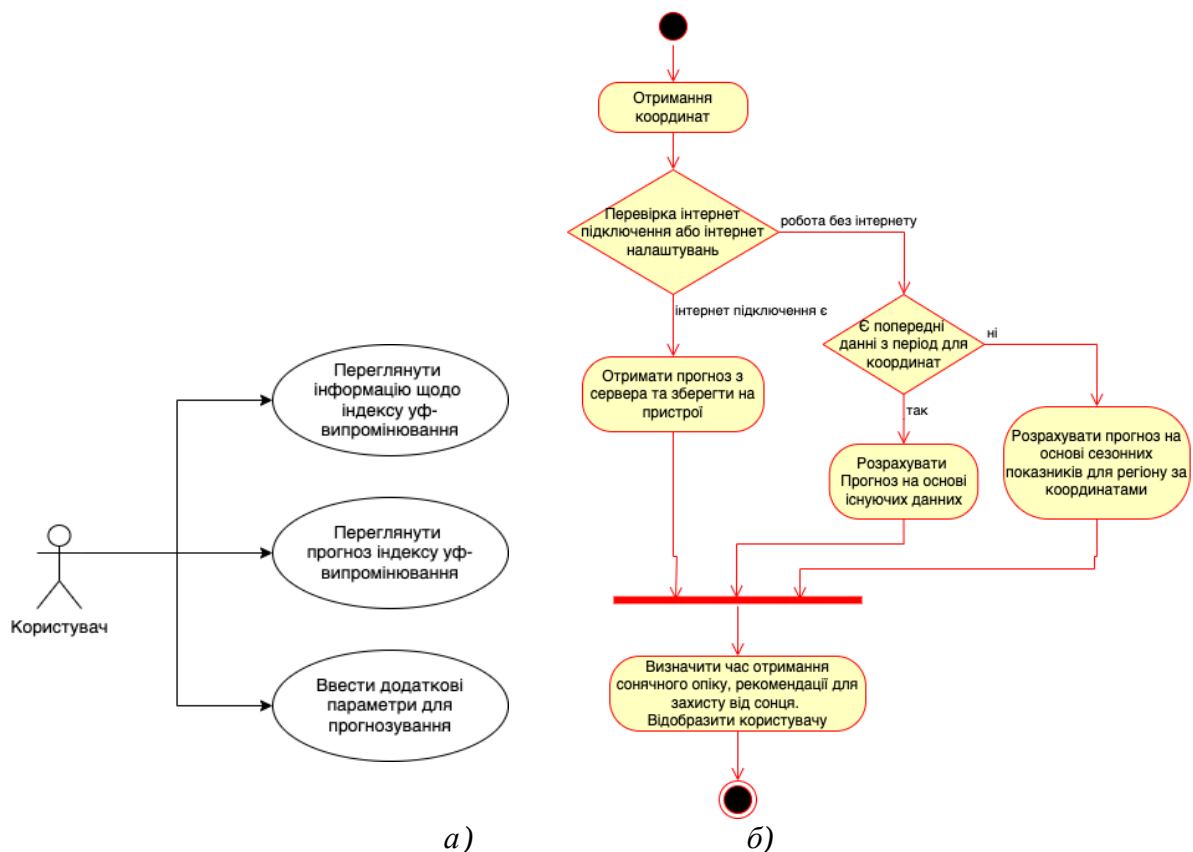


Рисунок 1 – Ескізний проєкт програмного забезпечення для оцінки впливу ультрафіолетового випромінювання екіпаж судна:

а) діаграма варіантів використання; б) діаграма діяльності

Застосунок не потребує глибокої декомпозиції, оскільки є достатньо простим та з мінімальної кількістю дій.

Алгоритм поведінки застосунку в умовах підключення до інтернету та його відсутності можна описати діаграмою послідовності дій, яка представлена на рис.1б.

Висновки 1. Представлений проєкт програмного забезпечення мобільного застосунку для оцінки впливу ультрафіолетового випромінювання на стан здоров'я екіпажу суден дозволить спрогнозувати ймовірний вплив уф-випромінювання на екіпаж судна в умовах обмеженості або відсутності зв'язку з урахуванням особливостей типу шкіри людини, її поточного міста знаходження та кліматичних умов. 2. Перспектива подальших досліджень полягає в удосконаленні математичної моделі розрахунку впливу ультрафіолетового випромінювання за рахунок розширення параметрів емпіричних даних.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Ultraviolet (UV) Radiation. (2017). <https://scied.ucar.edu/learning-zone/atmosphere/ultraviolet-uv-radiation>.
- [2] Health Effects of UV Radiation. (2022). Retrieved from: <https://www.epa.gov/sunsafety/health-effects-uv-radiation>.
- [3] Cancer Facts & Figures 2022. American Cancer Society. (2022). <https://www.cancer.org/content/dam/cancer-org/research/cancer-facts-and-statistics/annual-cancer-facts-and-figures/2022/2022-cancer-facts-and-figures.pdf>.
- [4] Sun-Protection and Sun-Exposure Habits Among Sailors: Results of the 2018 World's Largest Sailing Race Barcolana' Skin Cancer Prevention Campaign – PubMed. (2010) *PubMed*. <https://pubmed.ncbi.nlm.nih.gov/31442352/>.
- [5] Grandahl, K., Eriksen, P. (2018). Measurements of Solar Ultraviolet Radiation Exposure at Work and at Leisure in Danish Workers. *Danish Meteorological Institute*. doi: 10.1111/php.12920.
- [6] Global Solar UVI. (2002). <https://www.icnirp.org/en/publications/article/global-solar-uvi-2002.html>
- [7] Thomas B. Fitzpatrick. (1988). The Validity and Practicality of Sun-Reactive Skin Types I Through VI. *Arch Dermatol*. 124(6), 869-71. doi: 10.1001/archderm.124.6.869.
- [8] Allart, M., Weele, M., Fortuin, P. & Kelder, H. (2004). An empirical model to predict the UV-index based on solar zenith angles and total ozone. *Meteorological Applications*. 11(1), 59-65. doi:10.1017/S1350482703001130.
- [9] The Ozone Layer Center for Science Education. (2018). <https://scied.ucar.edu/learning-zone/atmosphere/ozone-layer>.

Farionova Tetyana Anatolyivna, Oriekhov Oleksandr Stanislavovich

Software project for assessing the impact of ultraviolet radiation on ship's crew

Abstract. Based on the analysis of the impact of ultraviolet radiation on the health of the ship's crew, existing measures to prevent the risk of damage to human skin, the need to develop software in the form of a mobile application is substantiated. The functionality is presented and the behavior algorithm of the software system is developed. The use of a mobile application will allow predicting the likely impact of UV radiation on the ship's crew in conditions of limited or no communication.

Keywords: mobile application; UV exposure assessment; ship crew; UV index.

АНАЛІЗ ІСНУЮЧИХ МАТЕМАТИЧНИХ МОДЕЛЕЙ ПРОГНОЗУВАННЯ ПРОДУКТИВНОСТІ ТА НАДІЙНОСТІ АКУМУЛЯТОРНИХ БАТАРЕЙ ЕЛЕКТРОМОБІЛЕЙ

Мешеряков В.І., д.т.н, проф¹; Устенко А.С.²

^{1,2} Одеський державний екологічний університет

^{1,2} Україна, Одеса

¹gradan@ua.fm; ²uas059877@gmail.com

Анотація. У даній роботі були розглянуті існуючі проблеми, які пов'язані із застосуванням акумуляторних батарей електромобілів. Наведено існуючі моделі прогнозування продуктивності акумуляторних батарей, а також їх деякі переваги та недоліки. Зроблено висновки щодо доцільності створення математичної моделі для підвищення експлуатаційної надійності акумуляторних батарей електромобілів. **Ключові слова:** надійність; акумуляторні батареї; електромобілі; моделювання.

Вступна частина. Сучасна енергетична промисловість переживає період переорієнтації на альтернативні види енергоресурсів натомість нафти, вугілля, газу. Це пов'язано з обмеженістю промислових видів сировини для палива та підвищенням вартості її видобутку. Водночас зростання населення планети, підвищення кількості екіпажів індивідуального користування призводить до значного підвищення вартості традиційного автомобільного палива - бензину. Серйозною проблемою сучасного світу є утворення суперміст, небувале збільшення населення в обмежених просторах. При цьому наявність великої кількості транспортних засобів призводить до серйозного, а іноді і небезпечного для життя забруднення довкілля. Ці обставини визначили підвищений інтерес до екологічно чистих альтернативних видів енергоносіїв для дорожньо-транспортних засобів, а саме акумуляторних батарей (АКБ) [1].

Основна частина. Застосування акумуляторних батарей надає певні переваги, такі як екологічність, порівняльна надійність та довговічність двигуна при тривалій експлуатації, можливість заряджання акумуляторів від стандартної електричної мережі, високий ККД. Незважаючи на це, виникають певні проблеми, вирішення яких дасть змогу підвищити надійність акумуляторних батарей:

- 1) поступова деградація батарей. У зв'язку зі зносом їх ємність падає, через що знижується запас ходу електромобілів.
- 2) вплив температурного режиму на експлуатацію батарей. Внаслідок загасаючої хімічної реакції, продуктивність АКБ при температурі нижче 0 °С значно знижується. При температурі вище 30 °С процес старіння АКБ прискорюється, а при температурі вище 40 °С акумулятор може бути пошкоджений.
- 3) необхідність у захисті батареї від ударів і вібрації. Батарея має бути ізольована відповідними матеріалами (наприклад, спінений поліпропілен), які б гасили вплив вібрації та несильних ударів. Для цього встановлюють сепаратори елементів батареї, які не тільки поглинають удари, а й запобігають деформації елементів унаслідок хімічних змін.
- 4) необхідність у захисті батареї від вологи. Треба застосовувати двоступінчасту вентиляцію для захисту відсіків та контейнерів батареї від вологи та забруднюючих речовин. Перший ступінь вирівнює тиск і запобігає попаданню води та забруднювачів. Другий ступінь вентиляції повністю відкривається у разі різкого підвищення тиску і нагріву, пропускаючи

гази, що розширюються, назовні і запобігаючи подальшому пошкодженню елементів, що залишилися.

Найбільш критичною для акумуляторних батарей є проблема терморегуляції. Виявилось, що в країнах із спекотним кліматом електромобілі почуваються найгірше. Те саме стосується частого використання швидких зарядних станцій — через високу потужність вони нагрівають батареї. У результаті електромобіль за такої експлуатації втрачає в середньому 10% за шість років.

Боротися з перегрівом батареї можна шляхом застосування замкнутої примусової рідинної системи охолодження. При перевищенні обумовленої температури батареї має включатися насос, який буде примушувати циркулювати охолоджуючу рідину. Рідина має рухатися по трубках, на яких, якщо це можливо, встановити ребра, хоча б невеликої довжини. Це сприятиме поліпшенню теплообмінних процесів.

Для реалізації цих заходів має бути встановлений мікропроцесор з датчиком температури. Для запобігання надмірному нагріву охолоджуючої рідини її також треба буде охолоджувати повітряним потоком у теплообмінному апараті. Подібна схема може працювати у місцевостях з помірним температурним режимом. Для країн з високим рівнем температури охолоджувач рідини має працювати за принципом кондиціонера. Тобто має бути встановленим компресор, а охолоджуючою рідиною виступатиме, наприклад фреон. Але це суттєво ускладнює систему охолодження акумуляторної батареї.

Батарея також має бути теплоізованою застосуванням таких матеріалів як пінополіпропілен та пінополістирол. Це трошки збільшить масу батареї, що не є критичним, але зменшить кліренс автомобілю. Для країн з поганими дорогами це буде критичним.

Перспективним для охолодження батарей автомобілів слід вважати застосування термоелектричних пристроїв. До основних переваг термоелектричного охолодження, як способу отримання штучного холоду, слід віднести зразкову відповідність показників надійності та масогабаритних характеристик термоелектричних пристроїв (ТЕП) та елементів електроніки простоті управління та швидкодії. Ці переваги за своєю сутністю є наслідком твердотільної природи охолоджувачів, тобто. відсутності рухомих частин, рідини або газів.

Як показав аналіз, незважаючи на велику кількість переваг, проблеми, що пов'язані із застосуванням акумуляторних батарей заважають масовому застосуванню електромобілей. Тому актуальною темою є підвищення надійності акумуляторних батарей з використанням математичного моделювання.

На даний час існують математичні моделі для прогнозування продуктивності АКБ. Головним їх призначенням є оцінювання ступеню деградації ємності АКБ в процесі її використання. Нижче наведено стислий перелік існуючих моделей АКБ:

- 1) Рівняння Батлера-Фермера. Ця модель призначена для прогнозування продуктивності батареї протягом часу. Її основним недоліком є те, що вона не враховує температурні режими, в якому вона працює.
- 2) Модель Томаса Фермі. Ця розширена модель вже враховує температурні режими, що дозволяє отримати більш точні результати, але ця модель вимагає значних обчислювальних ресурсів.
- 3) Моделі, що засновані на машинному навчанні. Такі моделі є найбільш сучасними та актуальними. Вони дозволяють отримувати найбільш ефективні результати з оцінювання максимального терміну роботи АКБ [2][3].

Висновки. Таким чином оцінювання надійності АКБ є актуальним та перспективним напрямом, що потребує розробки математичної моделі.

Для досягнення поставленої мети необхідно проаналізувати вхідні дані по кожному способу вирішення проблеми, побудувати математичну модель, перевірити її адекватність та створити програмне забезпечення, яке за отриманими даними розраховуватиме коефіцієнти даної моделі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Двигатель электромобиля, гибридного авто. <https://pro-sensys.com/info/articles/obzornye-stati/dvigatel-elektromobilya/> (2020, December 3)
- [2] Attia, P.M., Grover, A., Jin, N. et al. (2020). *Closed-loop optimization of fast-charging protocols for batteries with machine learning*. Nature 578, 397–402 <https://doi.org/10.1038/s41586-020-1994-5>
- [3] Dave, A., Mitchell, J., Burke, S. et al. (2022). *Autonomous optimization of non-aqueous Li-ion battery electrolytes via robotic experimentation and machine learning coupling*. Nat Commun 13, 5454 <https://doi.org/10.1038/s41467-022-32938-1>

Mesheryakov V.I., Ustenko A.S.

ANALYSIS OF EXISTING MATHEMATICAL MODELS FOR PREDICTION OF PERFORMANCE AND RELIABILITY OF ELECTRIC CAR BATTERIES

Abstract. *In this work, the existing problems associated with the use of electric car batteries were considered. The existing models for predicting battery performance are given, as well as some of their advantages and disadvantages. Conclusions were made regarding the expediency of creating a mathematical model for increasing the operational reliability of electric vehicle batteries*

Keywords: *reliability; battery system; electric vehicle, modelling*

УДК 519.254

КОНЦЕПЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ОБРОБКИ СТАТИСТИЧНИХ ДАНИХ НАУКОВИХ ДОСЛІДЖЕНЬ

Новожилова О. В., магістрант

Національний університет кораблебудування імені адмірала Макарова

Україна, Миколаїв

2151413@nuos.edu.ua

Анотація. Робота присвячена аналізу можливостей сучасних статистичних пакетів обробки даних та вирішенню науково-прикладного завдання створення інформаційного забезпечення обробки результатів статистичних даних, отриманих при проведенні наукових експериментів з подальшим використанням в науково-дослідницькій діяльності підприємств та учбових процесах.

Ключові слова: комп'ютерні статистичні пакети; статистичний аналіз даних; статистичні моделі; програмне забезпечення.

Вступна частина. Обробка статистичних даних полягає у визначенні значень числових характеристик, таких як математичне сподівання, дисперсія, асиметрія та ін. На основі отриманих значень характеристик формуються висновки щодо процесу дослідження. Такі розрахунки є досить трудомісткими і при виконанні вручну не гарантують необхідного рівня достовірності. Тому статистичну обробку дослідних даних доцільно виконувати за допомогою спеціалізованих прикладних програм для наукових розрахунків (MAXQDA, Mathcad 15.0, COSMOL Multiphysics, Wolfram Mathematica, Statgraphics, Statistica) [1,2].

Програмне забезпечення (ПЗ) для наукових розрахунків представлене багатьма організаціями: невеликими компаніями, корпораціями світового рівня. Спеціалізовані програми мають різне направлення - математика, астрономія, хімія, фізика, біологія, лінгвістика, інженерія, розробка штучного інтелекту тощо. ПЗ необхідне для відтворення графіків, обробки великих масивів інформації, зібраної науковими приладами. Згідно опитування SSI (Software Sustainability Institute), близько 92% вчених відповіли, що використовують у своїй практиці програми для наукових розрахунків. Більше половини з них розробляють власні програми, не маючи необхідних для цього навичок. А 69% опитаних висловили думку, що сучасні наукові дослідження просто неможливі без застосування комп'ютерної техніки і наукового софту [2].

У зв'язку з великим попитом статистичного аналізу даних практично в усіх сферах діяльності, особливо в науковій, ринок ПЗ для статистичної обробки даних швидко розвивається. Останнім часом отримали широке поширення програмні засоби або інформаційні системи, призначені для автоматизації робіт статистичної обробки даних, які дозволяють збирати, зберігати і обробляти різноманітні масиви даних з використанням єдиної інформаційної бази. Такі системи на підприємстві орієнтуються на потреби керівництва при виконанні функцій управління на основі внутрішніх і зовнішніх статистичних даних. Перевагами таких систем є адаптація інформаційної бази і функцій системи до умов функціонування підприємства. Проте, в силу складності реалізації математичних методів, такі системи, як правило, включають лише обмежений набір аналітичних методів.

Основна частина. При всьому багатстві програмного забезпечення статистичної обробки даних, етапи рішення статистичних завдань залишаються незмінними - побудова статистичної моделі, що відповідає завданню, вибір статистичного методу (разом з

програмою) і відповідного ПЗ, потім - реалізація цього методу в конкретному ПЗ для вирішення завдання. Отримання результату, що змістовно інтерпретується, не завжди означає закінчення дослідження. При використанні відібраних методів і засобів для аналізу інформації може бути потрібна безперервна робота впродовж декількох років. Використання ПЗ планується на усіх етапах дослідження, починаючи від організації зберігання даних і закінчуючи забезпеченням багатократного прогону завдань з різними параметрами.

Загальну логічну схему статистичного аналізу даних можна представити у вигляді семи етапів: початковий (попередній) аналіз досліджуваної системи; складання плану збору початкової інформації; збір початкових даних, їх підготовка і введення в ЕОМ; первинна статистична обробка даних; вибір основних методів і алгоритмів статистичної обробки даних, складання детального плану обчислювального аналізу матеріалу; реалізація плану обчислювального аналізу початкових даних (безпосередня експлуатація ЕОМ); підведення підсумків (будується формальний звіт про проведене дослідження, інтерпретуються результати застосування статистичних процедур) [3].

Проаналізувавши опубліковані результати досліджень, присвячених методам обробки статистичних даних, можна стверджувати, що більшість результатів вимірювання представляють собою вибірку з деякої генеральної сукупності. Особливо поширеними статистичними методами є: кореляційний аналіз, регресійний аналіз та факторний аналіз.[4]

Кореляційний аналіз. Між змінними (випадковими величинами) може існувати функціональний зв'язок, що проявляється в тому, що одна з них визначається як функція від іншої. Але між змінними може існувати і зв'язок іншого роду, що проявляється в тому, що одна з них реагує на зміну іншої зміною свого закону розподілу. Такий зв'язок називають стохастичним.

У регресійному аналізі моделюється взаємозв'язок однієї випадкової змінної від однієї або декількох інших випадкових змінних. При цьому перша змінна залежна, а інші - незалежні. Вибір або призначення залежної і незалежних змінних є довільним (умовним) і здійснюється дослідником залежно від вирішуваної їм задачі.

Факторний аналіз - один з найбільш популярних багатовимірних статистичних методів, він класифікує ознаки (змінні), що описують спостереження. Тому головна мета факторного аналізу - скорочення числа змінних на основі класифікації змінних і визначення структури взаємозв'язків між ними. Скорочення досягається шляхом виділення прихованих (латентних) загальних чинників, що пояснюють зв'язки між спостережуваними ознаками об'єкту, тобто замість початкового набору змінних з'явиться можливість аналізувати дані по виділених чинниках, число яких значно менше початкового числа взаємозв'язаних змінних.

Первинні дані, отримані при спостереженні, зазвичай важко інтерпретувати. Для того, щоб почати аналіз, в них потрібно внести деякий порядок і надати їм зручний для дослідника вигляд. Зокрема, спочатку бажано отримати уявлення про одновимірні розподіли випадкових величин, що входять в дані.

Візуалізація - це інструментарій, що дозволяє побачити кінцевий результат обчислень, організувати управління обчислювальним процесом і навіть повернутися назад до початкових даних, щоб визначити найбільш раціональний напрям подальшого руху.

Висновки. Проаналізувавши можливості та методи обробки статистичних даних із застосуванням спеціалізованих пакетів, можна зробити висновки, що вибір програмного забезпечення обробки статистичних даних - дуже складне питання. Для застосування ПЗ статистичних методів обробки даних користувачу не потрібно заглиблюватися в складність математичних процедур, а лише розуміти як ці методи використовуються. Для того, щоб

статистичний пакет програм був зручним та ефективним в роботі, він повинен задовольняти багатьом вимогам: містити значний спектр статистичного інструментарію (процедури, методи), відповідати вимогам введення, перетворення, збереження наборів даних, мати можливість експорту-імпорту даних, мати набір засобів візуалізації отриманих результатів, мати методичне забезпечення для самостійного освоєння користувачами, бути доступним за вартістю.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Фетісов, В.С. (2018). *Пакет статистичного аналізу даних STATISTICA*. Ніжин: НДУ імені М. Гоголя.
- [2] *Програми для наукових розрахунків*. (н.д) <http://softlist.com.ua/catalog/programmy-dlya-nauchnykh-raschetov/> (18.10.2022).
- [3] Осипенко, Н.Б., & Осипенко, А.Н. (2015). *Программные средства первичной статистической обработки экспериментальных данных: практическое руководство*. Гомель: ГГУ им. Ф. Скорины.
- [4] Грицюк, П.М., & Остапчук, О.П. (2008). *Аналіз даних: Навчальний посібник*. Рівне: НУВГП.

Novozhilova O.

Concept of the information system for processing statistical data of scientific research

Abstract. *The work is devoted to the analysis of the capabilities of modern statistical data processing packages and the solution of the scientific and applied task of creating information support for processing the results of statistical data obtained during scientific experiments with further use in the scientific and research activities of enterprises and educational processes.*

Keywords: *computer statistical packages; statistical data analysis; statistical models; software.*

УДК 005.8

ДОСЛІДЖЕННЯ МЕТОДІВ ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ НАВЧАННЯ

Гайдаєнко О.В., к.т.н., доцент¹; Бандурко К.Ю. магістр²; Світенко І.В. магістр³

^{1,2,3} Національний університет кораблебудування імені адмірала Макарова

^{1,2,3} Україна, Миколаїв

¹ ORCID 0000-0002-6614-5443; ² ekaterina-homenk@ukr.net; ³ igorsvitenko5@gmail.com

Анотація. Бурхливий розвиток систем дистанційного навчання, впровадження цифрових технологій у шкільний навчальний процес призвело до появи великих масивів освітніх даних. Розглянуто методи аналізу освітніх даних та їх ексклюзивність у порівнянні з методами інтелектуального аналізу даних.

Ключові слова: освітні дані; аналіз; масив даних.

Вступна частина. Широке поширення систем дистанційного навчання надає дослідникам значно більший обсяг інформації порівняно із традиційним освітнім процесом. Пов'язано це як з активним використанням в електронному навчанні різних технологій збору даних, так і з більшим масштабом аудиторії електронних освітніх середовищ. Зростання обсягу даних сприяло появі в початку 2000-х років нового напрямку в галузі штучного інтелекту – аналізу освітніх даних [1]. Якість освіти розглядається як комплексний процес на який впливає велика кількість факторів. Отримання якісної освіти безпосередньо залежить від якості самих вимог (цілей, стандартів і норм), якості ресурсів (програми, кадровий потенціал, контингент абітурієнтів, матеріально-технічне забезпечення, фінанси тощо) та якості освітніх процесів (наукова та навчальна діяльність, управління, освітні технології), які безпосередньо забезпечують підготовку фахівців. Методи аналізу корисні для проектування освітнього середовища, в організації матеріалів навчальних курсів та управління освітніми ресурсами.

Метою роботи є підвищення ефективності навчання за рахунок прийняття управлінських рішень, які ґрунтуються на більш глибокому розумінні процесів і закономірностей присутніх в освітньому процесі.

Основна частина. В освітньому аналізі даних широко використовуються методи традиційного аналізу, основні із яких є: класифікація, кластеризація, пошук зв'язуючих правил, пошук послідовних шаблонів, а також інтелектуальний аналіз текстів (текстовий аналіз). В [2] показано який метод і в яких саме завданнях частіше всього використовується.

Перевага кластерного аналізу в тому, що він дозволяє проводити розбиття об'єктів не по одному параметру, а по цілому набору ознак. Крім того, кластерний аналіз, на відміну від більшості математико-статистичних методів, не накладає ніяких обмежень на вид розглянутих об'єктів і дозволяє розглядати безліч вихідних даних практично довільної природи.

Кластерний аналіз дозволяє розглядати досить великий обсяг інформації і різко скорочувати, стискати великі масиви інформації, робити їх компактними і наочними.

Формально задача кластеризації описується наступним чином.

Для множини об'єктів даних I , кожний з яких представлено набором атрибутів. Необхідно побудувати множину кластерів C і відображення F множини I на множину C , тобто $F : I \rightarrow C$. Відображення F задає модель даних, яка є рішенням задачі. Якість рішення задачі визначається кількістю правильно класифікованих об'єктів даних.

Множину I визначаємо наступним чином:

$$I = \{i_1, i_2, \dots, i_j, \dots, i_n\},$$

де i_j – досліджуємий об'єкт.

Прикладом такої множини може служити набір даних про оцінки по дисциплінам.

Кожний з об'єктів характеризується набором параметрів:

$$i_j = \{x_1, x_2, \dots, x_h, \dots, x_m\}.$$

В нашому прикладі такими параметрами є питома вага лекцій, практичних, індивідуальних і лабораторних занять у загальній кількості кредитних годин виділених на дисципліну.

Кожна змінна x_h може приймати значення з деякої множини:

$$x_h = \{v_h^1, v_h^2, \dots\}.$$

В даному прикладі значеннями є дійсні числа.

Задача кластеризації полягає в побудові множини:

$$C = \{c_1, c_2, \dots, c_k, \dots, c_g\}.$$

Тут c_k – кластер, містить схожі один на одного об'єкти з множини I :

$$c_k = \{i_j, i_p \mid i_j \in I \cup d(i_j, i_p) < \sigma\},$$

де σ – величина, яка визначає міру близькості для включення об'єктів в один кластер; $d(i_j, i_p)$ – міра близькості між об'єктами, яка називається відстанню.

Невідомі значення $d(i_j, i_p)$ називаються відстанню між елементами i_j і i_p , якщо виконуються наступні умови:

- a) $d(i_j, i_p) \geq 0$, для всіх i_j і i_p ;
- b) $d(i_j, i_p) = 0$, тоді і тільки тоді, коли $i_j = i_p$;
- c) $d(i_j, i_p) = d(i_p, i_j)$;
- d) $d(i_j, i_p) \leq d(i_j, i_r) + d(i_r, i_p)$.

В останній час ведуться активні розробки нових алгоритмів кластеризації, які поєднують в собі методи ієрархічної кластеризації з іншими методами. Одним з таких алгоритмів є алгоритм BIRCH (Balanced Interactive Reducing and Clustering using Hierarchies) [3].

В ньому реалізовано двоетапний процес кластеризації.

В ході першого етапу формується попередній набір кластерів. На другому етапі до виявлених кластерам застосовуються інші алгоритми кластеризації – придатні для роботи в оперативній пам'яті.

Висновки. В результаті дослідження було прийнято рішення використовувати методи кластеризації та дерев рішень для подальшого їх використання у системі аналізу освітніх даних.

Реалізація даної системи дозволить підвищити ефективність навчання за рахунок прийняття управлінських рішень, які ґрунтуються на більш глибокому розумінні процесів і закономірностей присутніх в освітньому процесі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

[1] Михалко, М.І., Гайдаєнко, О.В. (2021). Автоматизована інформаційна система контролю та адміністрування відвідуваності університету. *Інформаційні технології: моделі, алгоритми, системи* :

- II Всеукраїнської науково-практичної інтернет конференції (ITMAS – 2021) (С. 63-65). Миколаїв, Україна. <http://itconf.nuos.edu.ua/2021/proceedings/>*
- [2] Romero, C., Ventura, S. (2010). Educational Data Mining: A Review of the State of the Art. *IEEE Transactions on Systems Man and Cybernetics Part C (Applications and Reviews)*. 010, V. 40, № 6. P. 601-618. doi: 10.1109/TSMCC.2010.2053532
- [3] Gupta, A. (2021, Jun 2). *Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH) Algorithm in Machine Learning*. <https://medium.com/geekculture/balanced-iterative-reducing-and-clustering-using-hierarchies-birch-1428bb06bb38>

Haydaenko O.V., Bandurko K.Yu., Svitenko I.V.

Research of methods of increasing the efficiency of learning

Abstract. *The rapid development of distance learning systems, the introduction of digital technologies into the school educational process has led to the emergence of large arrays of educational data. Methods of educational data analysis and their exclusivity in comparison with methods of intellectual data analysis are considered.*

Keywords: *educational data; analysis; array of data.*

УДК 519.722

ВИКОРИСТАННЯ ЕНТРОПІЇ ДЛЯ ЗНАХОДЖЕННЯ ВНЕСЕННЯ ЗМІН КОДУ ВИКОНУВАНИХ ФАЙЛІВ

Макарова Л.М., к.т.н, доцент¹; Камінський С.С.²; Бризгалов М.В.³

^{1,2,3} Національний університет кораблебудування імені адмірала Макарова

^{1,2,3} Україна, Миколаїв

¹lidiia.makarova@nuos.edu.ua; ²191181@nuos.edu.ua; ³201166@nuos.edu.ua

Анотація. Розроблено програму для обчислення ентропії дизасембльованих виконуваних файлів і порівняння з минулими значеннями з метою виявлення авторства зміни коду виконуваних файлів.

Ключові слова: ентропія, дизасемблювання, виконувані файли, зміна коду, програмний додаток.

Вступна частина. Розвиток сучасних комп'ютерних технологій породжує безліч нових можливостей, оптимізації звичайних процесів та появ нових перспектив. В основі кожного процесу лежить передача даних і виконання коду на комп'ютері, які ми використовуємо щодня і не замислюємося про всю комплексність процесів, що проходять. Чим більшими і складнішими стають ці системи, тим вище стають вимоги до безпеки на кожному етапі і тим більше потенційних слабкостей у них виявляється. Однією з них є ін'єкція коду у виконуваний файл. Саме такі тривіальні речі можуть призвести до величезних втрат, потенційної небезпеки втрати важливих даних та їхнього потрапляння до третіх осіб [1].

Одним із способів вирішення цієї проблеми є обчислення метрик вихідного або виконаного програмного коду. Було розглянуто різні метрики програмного забезпечення з точки зору їх застосування для визначення авторства програмного коду, отже, до виявлення впровадження чужорідного коду.

Так, кількісна метрика – ентропія – добре підходить для визначення авторства програмного коду, тобто, використовуючи дану метрику, можна з великою ймовірністю стверджувати, написана нова ділянка коду автором програми чи ні [2].

Метою роботи є розробка програми для розрахунку ентропії дизасембльованих виконуваних файлів та порівняння з минулими значеннями з ціллю виявлення авторства зміни коду виконуваних файлів.

Для того, щоб досягти поставленої мети потрібно вирішити наступні завдання:

- проаналізувати існуючі методи дизасемблювання та відповідні програми;
- знайти дизасемблер для виконуваних файлів, який задовольняє заданим вимогам;
- розробити програму для аналізу виконуваних файлів та їх дизасембльованого коду;
- розрахувати метрику ентропії для дизасембльованого коду;

- провести аналіз отриманих даних і зробити висновки, базуючись на ентропії, на предмет внесення авторських чи чужорідних змін.

Основна частина. Основна складність полягає у принципі аналізу виконуваних файлів щодо знаходження чужорідного коду. Для цього будемо використовувати значення метрики ентропії.

Для аналізу коду використовується розрахунок ентропії, що дозволяє проаналізувати файли та відрізняти їх по результатам обчислення.

Інформаційна ентропія – міра невизначеності або непередбачуваності інформації, невизначеність появи будь-якого символу первинного алфавіту [3]. За відсутності інформаційних втрат ентропія чисельна дорівнює кількості інформації на символ переданого повідомлення. Наприклад, у послідовності букв, що складають будь-яке речення українською мовою, різні букви з'являються з різною частотою, тому невизначеність появи для деяких букв менша, ніж для інших. Якщо врахувати, що деякі поєднання букв зустрічаються дуже рідко, то невизначеність зменшується ще сильніше.

Ентропія також може бути застосована до програмного забезпечення, як міра взаємодії. Так, розглядаючи ентропію вихідного коду, який може бути розбитий на більш дрібні сегменти (рядки, функції, змінні, команди і т.п.), можна визначати авторство тієї чи іншої частини коду. Таким чином, можна обчислити внесок різних авторів у розроблену програму. Для розрахунку ентропії в теорії інформації використовують таку формулу [4]:

$$H(i) = -p_i * \log_2(p_i), \quad (1)$$

де i - можливі стани;

p_i - імовірність появи i -го стану.

У випадку з програмним кодом у ролі станів виступають команди, і, відповідно, підраховується ймовірність їх появи в коді. Деякі метрики програмного забезпечення, наприклад, метрики Холстеда чи Берлінгера, беруть за основу ентропію. Також ентропійний метод використовується для вирішення таких завдань, як пошук зашифрованого або запакованого шкідливого програмного забезпечення.

Алгоритм роботи програми:

1) Дизасемблювання виконуваного файлу. Дизасемблювання здійснювалося за допомогою утиліти dumpbin.exe, яка входить до складу Microsoft Visual Studio [5]. В результаті отримуємо txt-файл, в якому знаходиться код програми.

2) Розрахунок значень ентропії кожної команди для отриманого дизасембльованого файлу. Обчислюємо за формулою (1). Усі отримані значення записуються у файл.

3) Розрахунок зміни ентропії. Зчитуємо значення ентропії для кожної команди оригінального файлу та відповідного йому зміненого. Підраховуємо різницю та записуємо результат у файл.

4) Визначення порогового значення ентропії кожної команди.

5) Порівняння різниці ентропії відповідних команд із пороговим значенням для цих команд.

6) Розрахунок критерію розбіжності.

Програма дозволяє дизасемблювати виконувані файли, тим самим отримати асемблерний код, розрахувати значення ентропії для кожної команди, проаналізувати отримані дані з точки зору різниці ентропії.

Програма зберігає зліпки файлів за найменуванням. Якщо якийсь файл був дизасембльований і буде завантажений новий з такою самою назвою, програма проведе порівняння між ними та відобразить результат порівняння.

Приклад роботи розробленої програми, а саме розрахунок ентропії для поточної та попередньої версії файлу, наведено на рисунку 1.



Рисунок 1 – Приклад розрахунку ентропії для поточної та попередньої версії файлу

Висновки. Була розроблена програма для аналізу дизасембльованого коду виконуваних файлів, обчислення ентропії та порівняння з минулими значеннями для знаходження змін коду виконуваних файлів. Розроблена програма може бути використана зокрема для валідації файлів існуючих програм. В подальшому передбачається удосконалення розробленої програми за рахунок додавання математичної моделі для прогнозування авторства внесених змін.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] Що таке ін'єкція коду в Windows? (2018, Серпень 27). <https://www.thefastcode.com/uk-ua/article/what-is-code-injection-on-windows>
- [2] Энтропия. Как хаос помогает искать вирусы. (2021, Січень 29). <https://haker.ru/2021/01/29/viruses-entropy/>
- [3] Формула Шеннона. (n.d.). <http://um.co.ua/8/8-16/8-168268.html>
- [4] Шарапов, О.Д., Дербенцев В.Д., & Семьонов, Д.С. (2004). Економічна кібернетика. Навч. посібник. К.: КНЕУ.
- [5] Microsoft Visual Studio disassembler “dumpbin”. (2022, Вересень 27). <https://learn.microsoft.com/ru-ru/cpp/build/reference/dumpbin-reference?view=msvc-170>

Makarova L.M., Kaminskyi S.S., Bryzghalov M.V.

The entropy metric for changes search in the code of executable files

Abstract. A program for calculating the entropy of disassembled executable files and comparing them with past values in order to identify the authorship of changes in the code was developed.

Keywords: entropy, disassembly, executable files, code change, software application.