

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут комп'ютерних наук та управління проектами
Кафедра програмного забезпечення автоматизованих систем
Спеціальність 121 «Інженерія програмного забезпечення»
Освітня програма «Інженерія програмного забезпечення»

«Допущений до захисту»
Завідувач кафедри
_____ проф. Приходько С.Б.

« ____ » _____ 2022 р.

КВАЛІФІКАЦІЙНА РОБОТА
на здобуття ступеня вищої освіти «магістр»

на тему: Удосконалення регресійної моделі для оцінювання метрики RFC застосунків з відкритим кодом на Java та розробка програми для її реалізації

Виконав: студент групи 6151м
_____ Позняков Р.О.
(підпис, ПБ)

Керівник роботи:
Завідувач кафедри, д.т.н., професор
(посада, науковий ступень вчене звання)
_____ Приходько С.Б.
(підпис, ПБ)

Миколаїв – 2022 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет кораблебудування імені адмірала Макарова
 Навчально-науковий інститут комп'ютерних наук та управління проектами
 Кафедра програмного забезпечення автоматизованих систем
 Спеціальність 121 «Інженерія програмного забезпечення»
 Освітня програма «Інженерія програмного забезпечення»

«ЗАТВЕРДЖУЮ»
 Гарант освітньої програми
 _____ проф. С.Б.Приходько
 (підпис)
 «10» жовтня 2022 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ
на здобуття ступеня вищої освіти «магістр»

Студенту Познякову Роману Олександровичу _____
 (Прізвище, ім'я, по батькові)

1. Тема роботи: Удосконалення регресійної моделі для оцінювання метрики RFC застосунків з відкритим кодом на Java та розробка програми для її реалізації

Керівник роботи Приходько С.Б. _____

Затверджені наказом ректора № 798 уч від «28» _____ 09 _____ 2022 р.

2. Термін подання роботи: 14.11.2022 р. _____

3. Вихідні дані по роботі: _____

4. Перелік питань, що належать до розробки (найменування розділів) _____

Вступ (Актуальність теми. Зв'язок роботи з науковими програмами, планами, темами. Мета і завдання дослідження. Об'єкт дослідження. Предмет дослідження. Методи дослідження. Наукова новизна одержаних результатів. Практичне значення одержаних результатів. Особистий внесок здобувача. Апробація результатів досліджень. Публікації.); Огляд літератури та обґрунтування необхідності проведення досліджень за обраною темою; Викладення результатів власних досліджень з висвітленням того нового, що пропонується; Проект програмного забезпечення; Організаційно-економічний розділ; Розділи з охорони праці та охорони навколишнього середовища; Висновки; Список використаних джерел; Додатки (технічне завдання, текст програми, опис програми, інструкція користувача, програма і методика випробувань програмного забезпечення)

5. Перелік презентаційних матеріалів: Актуальність теми. Мета і завдання дослідження. Об'єкт дослідження. Предмет дослідження. Методи дослідження. Наукова новизна одержаних результатів. Практичне значення одержаних результатів. Особистий внесок здобувача. Апробація результатів досліджень. Публікації. Існуючі регресійні моделі для оцінювання метрики RFC. Нелінійна регресійна модель для оцінювання метрики RFC застосунків з відкритим кодом на Java. Постановка задачі на розробку проекту програми. Діаграма варіантів використання. Діаграма діяльності. Графічне вікно програми із інтерфейсними компонентами. Емпіричні дані. Результати випробувань програми. Висновки. _____

РЕФЕРАТ

Позняков Роман Олександрович

«Удосконалення регресійної моделі для оцінювання метрики RFC застосунків з відкритим кодом на Java та розробка програми для її реалізації»

Кваліфікаційна робота на здобуття освітнього рівня магістра зі спеціальності 121 – «Інженерія програмного забезпечення». Національний університет кораблебудування імені адмірала Макарова. Миколаїв, 2022 р.

Обсяг роботи: 108 стор., 6 табл., 8 рис., 19 використаних джерел, 5 додатків.

Актуальність теми роботи: необхідність підвищення достовірності оцінювання метрики RFC застосунків з відкритим кодом на Java.

Мета та завдання дослідження. Метою є підвищення достовірності оцінювання метрики RFC застосунків з відкритим кодом на Java. Завдання дослідження: проаналізувати існуючі регресійні моделі для оцінювання метрики RFC застосунків на Java; удосконалити регресійну модель для оцінювання метрики RFC застосунків з відкритим кодом на Java; розробити програму для оцінювання метрики RFC застосунків з відкритим кодом на Java.

Об'єктом дослідження є процес оцінювання метрики RFC застосунків з відкритим кодом на Java. **Предметом дослідження** є регресійні моделі для оцінювання метрики RFC застосунків з відкритим кодом на Java.

Методи дослідження. У вирішенні поставлених завдань використано методи теорії ймовірності, математичної статистики, регресійного аналізу.

Наукова новизна одержаних результатів. Удосконалено регресійну модель для оцінювання метрики RFC застосунків з відкритим кодом на Java на основі нормалізуючого перетворення у формі десяткового логарифму, яка дозволяє підвищити достовірність оцінювання метрики RFC застосунків з відкритим кодом на Java, в порівнянні з існуючими регресійними моделями.

Практичне значення одержаних результатів. Розроблено програму для оцінювання метрики RFC застосунків з відкритим кодом на Java для системи моделювання Scilab 6.0.0.

Апробація результатів досліджень. Результати досліджень були оприлюднені на Всеукраїнській науково-практичній інтернет-конференції.

Публікації. Результати роботи викладені у 1 публікації, а саме: 1 матеріалах Всеукраїнської науково-практичної інтернет-конференції.

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

ABSTRACT

Roman Oleksandrovich Poznyakov

«Improving the regression model to estimate the RFC metric of open-source applications in Java and developing the software for its implementation»

The qualification work in obtaining a master's degree in specialty 121 - "Software Engineering". Admiral Makarov National University of Shipbuilding. Mykolayiv, 2022.

The qualification work is presented on the 108 pages of typewritten text, contains 6 tables, 8 figures, 5 appendices and 19 references.

Relevance of the topic of the work: the need to increase the confidence of estimating the RFC metric of open-source applications in Java.

The goal and objectives of the study. The goal of the study is to increase the confidence of estimating the RFC metric of open-source applications in Java by nonlinear regression model. The objectives of the study: to analyze existing regression models to estimate the RFC metric of applications in Java; to improve the regression model for estimating the RFC metric of open-source applications in Java; to develop the software for estimating the RFC metric of open-source applications in Java.

The object of the study is the process of estimating the RFC metric of open-source applications in Java.

The subject of the study is the regression models for estimating the RFC metric of open-source applications in Java.

The methods of the study. The methods of probability theory, mathematical statistics, regression analysis are used in solving the problems.

The scientific novelty of obtained results. The nonlinear regression model for estimating the RFC metric of open-source applications in Java is improved based on normalizing transformation in the decimal logarithm form which allows to increase confidence of estimating the RFC metric of open-source applications in Java in comparison with existing linear regression models.

The practical significance of obtained results is that the software realizing the constructed model is developed for the simulation system of Scilab 6.0.0.

Approbation of study results. The study results were presented at the All-Ukrainian scientific Internet-conference.

Publications. The study results were published in one publication: the materials of the All-Ukrainian scientific Internet-conference.

Keywords: regression model, estimation, the RFC metric, open-source application, Java.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ	8
ВСТУП	9
1 АНАЛІЗ ІСНУЮЧИХ МОДЕЛЕЙ ДЛЯ ОЦІНЮВАННЯ МЕТРИКИ RFC ЗАСТОСУНКІВ НА JAVA ТА ОБҐРУНТУВАННЯ НЕОБХІДНОСТІ ПРОВЕДЕННЯ ДОСЛІДЖЕНЬ ЗА ОБРАНОЮ ТЕМОЮ	13
1.1 Аналіз існуючих моделей для оцінювання метрики RFC застосунків на Java	13
1.2 Обґрунтування необхідності проведення досліджень за обраною темою	15
1.3 Висновки за розділом 1	21
2 НЕЛІНІЙНА РЕГРЕСІЙНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ МЕТРИКИ RFC ЗАСТОСУНКІВ З ВІДКРИТИМ КОДОМ НА JAVA	23
2.1 Перевірка даних для побудови нелінійної регресійної моделі для оцінювання метрики RFC застосунків з відкритим кодом на Java на наявність викидів	23
2.2 Побудова нелінійної регресійної моделі для оцінювання метрики RFC застосунків з відкритим кодом на Java	24
2.3 Висновки за розділом 2	32
3 РОЗРОБКА ПРОЕКТУ ПРОГРАМИ ДЛЯ ОЦІНЮВАННЯ МЕТРИКИ RFC ЗАСТОСУНКІВ З ВІДКРИТИМ КОДОМ НА JAVA	34

3.1	Постановка задачі на розробку проекту програми для оцінювання метрики RFC застосунків з відкритим кодом на Java	34
3.2	Розробка ескізного проекту програми для оцінювання метрики RFC застосунків з відкритим кодом на Java	36
3.3	Розробка технічного проекту програми для оцінювання метрики RFC застосунків з відкритим кодом на Java	42
3.4	Розробка робочого проекту програми для оцінювання метрики RFC застосунків з відкритим кодом на Java	46
3.5	Висновки за розділом 3	55
4	РОЗРАХУНОК ЕКОНОМІЧНОЇ ЕФЕКТИВНОСТІ ВПРОВАДЖЕННЯ ПРОГРАМИ	56
4.1	Розрахунок витрат на створення й експлуатацію програми	56
4.2	Економічна ефективність розробки і впровадження програми	59
5	ОХОРОНА ПРАЦІ	62
5.1	Аналіз існуючої проблеми охорони праці	62
5.2	Аналіз шкідливих та небезпечних факторів в офісі фірми з персональними комп'ютерами	63
5.3	Розрахунок рівня освітлення в офісі фірми	70
5.4	Розробка заходів щодо зменшення впливу шкідливих та небезпечних факторів	72
6	ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА	75
6.1	Аналіз існуючої проблеми необхідності охорони навколишнього середовища	75
6.2	Утилізація відходів комп'ютерної техніки	78
	ВИСНОВКИ	83
	СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	85

ДОДАТОК А Технічне завдання на розробку програми для оцінювання метрики RFC застосунків з відкритим кодом на Java	89
ДОДАТОК Б Текст програми	93
ДОДАТОК В Опис програми	100
ДОДАТОК Г Інструкція користувача	101
ДОДАТОК Д Програма та методика випробувань	103

**ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,
СКОРОЧЕНЬ І ТЕРМІНІВ**

ПЗ – програмне забезпечення;

CBO – Coupling Between Objects classes;

LB – lower bound;

MD – the Mahalanobis distance;

MMRE – a mean magnitude of relative error;

MRE – a magnitude of relative error;

OOD – object-oriented design;

PRED – percentage of prediction;

RFC – the Response for Class;

SMD – the squared Mahalanobis distance;

UB – upper bound;

WMC – weighted methods per class.

ВСТУП

Актуальність теми. 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, 2]. Однак відомі лише точкові оцінки їх середніх для визначення якості програмних систем [1]. Наприклад, за даними [1] програмна система має високу якість у разі, якщо середні значення її метрик RFC, CBO та WMC є відповідно такими: 43,84, 1,25 та 11,10. А інтервальних оцінок для показників ПЗ на рівні програми для визначення її якості поки не знайдено.

Також відомо [1, 2], що метрика RFC залежить від метрик CBO та WMC. У [1] автори побудували рівняння лінійної регресії для оцінювання середнього RFC в залежності від метрик CBO та WMC. Водночас не завжди існує теоретичне обґрунтування використання моделей лінійної регресії, наприклад, коли розподіл залишків не є гаусовим [3]. У цьому випадку необхідно застосовувати нелінійні регресійні моделі [4-8]. Зазначене вказує на актуальність теми.

Мета і завдання дослідження. Метою роботи є підвищення достовірності оцінювання метрики RFC в залежності від метрик CBO та WMC застосунків з відкритим кодом на Java для визначення їх якості.

Для досягнення поставленої мети потрібно вирішити такі завдання.

- Проаналізувати існуючі регресійні моделі для оцінювання метрики RFC в залежності від метрик CBO та WMC застосунків з відкритим кодом на Java.

- Здійснити підбор сукупності даних, за якими буде будуватися нелінійна регресійна модель для оцінювання метрики RFC в залежності від метрик CBO та WMC застосунків з відкритим кодом на Java та перевірити ці дані на наявність багатовимірних викидів.

- Удосконалити нелінійну регресійну модель для оцінювання метрики RFC в залежності від метрик CBO та WMC застосунків з відкритим кодом на Java на основі нормалізуючого перетворення.

- На підставі запропонованої нелінійної регресійної моделі розробити програму для оцінювання метрики RFC в залежності від метрик CBO та WMC застосунків з відкритим кодом на Java.

Об'єктом дослідження є процес оцінювання метрики RFC в залежності від метрик CBO та WMC застосунків з відкритим кодом на Java.

Предметом дослідження є регресійні моделі для оцінювання метрики RFC в залежності від метрик CBO та WMC застосунків з відкритим кодом на Java.

Методи дослідження. У вирішенні поставлених завдань використано методи теорії ймовірності, математичної статистики, регресійного аналізу.

Наукова новизна одержаних результатів. Удосконалено двофакторну нелінійну регресійну модель для оцінювання метрики RFC в залежності від метрик CBO та WMC застосунків з відкритим кодом на Java на основі нормалізуючого перетворення у формі десяткового логарифму із врахуванням викидів у регресії. Ця нелінійна регресійна модель у порівнянні з існуючими регресійними моделями, що побудовані без врахуванням викидів у регресії, має

більші значення множинного коефіцієнту детермінації і відсотка прогнозування, менші значення середньої величини відносної похибки та ширин довірчого інтервалу і інтервалу прогнозування для більшої кількості значень метрик СВО та WMC.

Практичне значення одержаних результатів. Розроблено програму для оцінювання метрики RFC в залежності від метрик СВО та WMC застосунків з відкритим кодом на Java. Програму створено sci-мовою для системи Scilab. Програму, що розроблено, можна використовувати для оцінювання якості відповідних застосунків.

Особистий внесок здобувача. У публікації [9], що зроблена у співавторстві, автору належить нелінійна регресійна модель для оцінювання метрики RFC в залежності від метрик СВО та WMC застосунків з відкритим кодом на Java на основі нормалізуючого перетворення у формі десяткового логарифму із врахуванням викидів у регресії, яка була побудована під керівництвом керівника кваліфікаційної роботи проф. Приходько С.Б. У кваліфікаційній роботі також використаний матеріал доповіді «A Technique for Detecting Software Quality Based on the Confidence and Prediction Intervals of Nonlinear Regression for RFC Metric», яка була зроблена проф. Приходько С.Б. на конференції «IEEE 17th International Conference on Computer Science and Information Technologies (CSIT2022)» за секцією «Software engineering» 11.11.2022 (<http://csit.ieee.org.ua/accepted-papers/#1631800275024-ee886edc-5296>).

Апробація результатів досліджень. Результати досліджень, викладені у роботі, були оприлюднені на одній конференції: III-ій Всеукраїнській науково-практичній інтернет-конференції «Інформаційні технології: моделі, алгоритми, системи (ITMAS – 2022)» (Миколаїв, 26-28 жовтня 2022 р.).

Публікації. За результатами кваліфікаційної роботи зроблено одну публікацію, а саме: матеріали Всеукраїнської науково-практичної інтернет-конференції.

1 АНАЛІЗ ІСНУЮЧИХ МОДЕЛЕЙ ДЛЯ ОЦІНЮВАННЯ МЕТРИКИ RFC ЗАСТОСУНКІВ НА JAVA, ТА ОБҐРУНТУВАННЯ НЕОБХІДНОСТІ ПРОВЕДЕННЯ ДОСЛІДЖЕНЬ ЗА ОБРАНОЮ ТЕМОЮ

1.1 Аналіз існуючих моделей для оцінювання метрики RFC застосунків на Java

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, 2]. Однак відомі лише точкові оцінки їх середніх для визначення якості програмних систем [1]. Наприклад, за даними [1] програмна система має високу якість у разі, якщо середні значення її метрик RFC, CBO та WMC є відповідно такими: 43,84, 1,25 та 11,10. А інтервальних оцінок для показників ПЗ на рівні програми для визначення її якості поки не знайдено.

Також відомо [1, 2], що метрика RFC залежить від метрик CBO та WMC. У [1] автори побудували рівняння лінійної регресії для оцінювання середнього RFC в залежності від метрик CBO та WMC

$$\hat{Y} = \hat{b}_0 + \hat{b}_1 X_1 + \hat{b}_2 X_2, \quad (1.1)$$

де $\hat{Y}$ – оцінка середнього метрики RFC за лінійним регресійним рівнянням (1.1); X_1 та X_2 – це значення метрик CBO та WMC; $\hat{b}_0$, $\hat{b}_1$ та $\hat{b}_2$ – це оцінки параметрів рівнянням (1.1).

Водночас не завжди існує теоретичне обґрунтування використання моделей лінійної регресії, наприклад, коли розподіл залишків не є гаусовим [3]. У цьому випадку необхідно застосовувати нелінійні регресійні моделі [4-8].

Лінійну регресійну модель у нашому випадку можна записати як

$$Y = \hat{Y} + \varepsilon = \hat{b}_0 + \hat{b}_1 X_1 + \hat{b}_2 X_2 + \varepsilon, \quad (1.2)$$

де ε — це випадкова величина, яка описує залишки, вона повинна бути випадковою величиною з розподілом Гаусу, $\varepsilon \sim N(0, \sigma_\varepsilon^2)$, з оцінкою дисперсії $\hat{\sigma}_\varepsilon^2$.

Крім того, навіть, якщо розподіл випадкової величини ε в (1.2) є гаусівським, то в більшості випадків лінійна регресійна модель має гірші показники якості у порівнянні нелінійними регресійними моделями [4-6].

Побудову нелінійної регресійної моделі для оцінювання метрики RFC застосунків з відкритим кодом на Java ми будемо за методом [7], що заснований на багатовимірних нормалізуючих перетвореннях та інтервалах прогнозування. Суть цього методу [7] є такою. На першому етапі початкові дані перевіряються на наявність викидів і, якщо останні знайдено, то вони відкидаються. Для цього використовується критерій на основі квадрату відстані Махаланобіса для нормалізованих даних із 0,005 рівнем значущості. На другому етапі будується нелінійна регресійна модель із використанням відповідного методу на основі нормалізуючих перетворень [6]. На третьому етапі визначаються границі інтервалу прогнозування нелінійної регресії для рівня значущості, що дорівнює 0,05, за відповідним методом [6]. На останньому четвертому етапі перевіряють, чи є серед даних, за якими будувалася нелінійна регресійна модель такі, що виходять за границі інтервалу прогнозування. Та, якщо відповідні дані

знайдено, то вони відкидаються, і ми повторюємо знову всі етапи, починаючи з першого, для нових даних. Якщо таких викидів не було, то повторення етапів завершується, відповідна нелінійна регресійна модель побудована.

1.2 Обґрунтування необхідності проведення досліджень за обраною темою

Як було зазначено в попередньому підрозділі, існує лінійне регресійне рівняння для оцінювання середнього значення метрики RFC в залежності від метрик CBO та WMC. Але виникає питання стосовно можливості застосування цього лінійного рівняння саме для метрик програмних застосунків з відкритим кодом на Java. Тому для остаточного обґрунтування необхідності проведення досліджень за обраною темою, перевіримо можливість використання зазначеної моделі для оцінювання метрики RFC в залежності від метрик CBO та WMC для застосунків з відкритим кодом на Java. 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>). Значення цих метрик для 49 застосунків наведені в табл.1.1. В табл.1.1 метрики RFC, CBO та WMC позначені відповідно як Y , X_1 та X_2 .

Далі ми перевірили багатовимірні дані метрик з табл.1.1 на нормальність за допомогою тесту Мардіа (Mardia's test) [10], якій базується на багатовимірних асиметрії та ексцесі. Тест показав, що розподіл даних не є гаусівським.

Таблиця 1.1

Набір даних та квадрат відстані Махаланобіса (MD)

№	Y	X ₁	X ₂	квадрат MD	
				без перетворення	з перетворенням
1	2	3	4	6	7
1	9,864	10,409	10,227	1,35	2,62
2	26,448	11,904	71,837	23,30	11,26
3	8,688	10,047	11,101	1,70	3,99
4	15,389	11,130	17,634	0,15	0,11
5	23,885	17,844	28,454	2,71	2,36
6	10,488	9,112	11,303	1,03	1,46
7	31,526	17,333	33,684	2,92	2,61
8	11,143	5,071	14,643	3,78	7,77
9	37,278	22,417	35,694	8,32	4,92
10	7,306	5,205	12,985	3,60	7,58
11	21,130	13,376	15,988	0,73	1,16
12	32,010	18,720	23,409	5,03	3,48
13	14,556	10,924	17,020	0,24	0,23
14	13,428	11,840	11,427	0,80	1,29
15	10,868	8,838	10,794	1,02	1,38
16	12,488	10,935	17,154	0,75	1,39
17	19,467	13,333	11,533	1,24	3,06
18	43,225	9,925	21,525	31,46	19,03
19	22,947	10,474	29,737	1,86	2,33
20	9,077	8,808	11,308	1,41	2,60
21	16,351	7,392	24,094	2,61	4,12
22	22,215	14,348	17,922	0,83	1,01
23	12,969	9,832	17,152	0,57	0,77
24	15,020	10,245	14,617	0,31	0,24
25	19,751	18,527	18,994	4,96	3,11

Продовження табл.1.1

1	2	3	4	6	7
26	14,125	8,625	22,250	1,32	2,02
27	24,214	15,963	25,237	1,10	1,16
28	14,922	11,912	18,546	0,36	0,56
29	12,653	11,056	14,722	0,63	0,87
30	18,124	11,970	6,932	1,97	9,67
31	9,521	10,490	8,042	1,59	3,76
32	25,931	16,028	25,014	1,33	1,31
33	16,265	11,529	11,803	0,52	1,37
34	18,629	10,077	15,593	0,88	1,30
35	37,158	17,814	65,838	13,45	7,69
36	12,974	9,103	20,253	1,02	1,68
37	12,596	10,626	14,063	0,56	0,69
38	24,936	17,387	29,284	2,12	2,06
39	15,586	9,590	15,554	0,50	0,47
40	25,388	13,800	42,113	3,25	3,50
41	18,966	16,269	23,796	2,28	2,17
42	16,735	9,559	15,397	0,70	0,88
43	20,526	14,792	18,032	0,84	0,81
44	16,370	6,111	15,926	3,94	7,61
45	10,622	7,519	12,949	1,53	1,90
46	13,100	10,529	13,743	0,46	0,47
47	15,241	12,693	11,274	0,89	1,69
48	15,329	12,744	11,382	0,88	1,65
49	22,707	17,240	25,216	2,21	1,84

Тому для визначення наявності викидів у даних ми використовували тест на основі квадрату відстані Махаланобіса (Mahalanobis Distance) для нормалізованих даних як це рекомендується робити у роботах [11, 12]. При

цьому, як рекомендується в [11-13], ми брали рівень значимості 0,005. Для нормалізації даних ми використовували десятковий логарифм.

Значення квадрату відстані Махаланобіса, яку далі позначаємо як MD, для кожної багатовимірної точки даних i , $i = 1, 2, \dots, N$, визначається як

$$d_i^2 = (\mathbf{Z}_i - \bar{\mathbf{Z}})^T S_N^{-1} (\mathbf{Z}_i - \bar{\mathbf{Z}}), \quad (1.3)$$

де $\bar{\mathbf{Z}}$ – вектор вибірових середніх; S_N – вибірова коваріаційна матриця

$$S_N = \frac{1}{N} \sum_{i=1}^N (\mathbf{Z}_i - \bar{\mathbf{Z}})(\mathbf{Z}_i - \bar{\mathbf{Z}})^T. \quad (1.4)$$

Тут $\mathbf{Z}_i$ – гаусівський випадковий вектор $\mathbf{Z} = \{Z_1, Z_2, \dots, Z_m\}^T$ для кожної багатовимірної точки даних i , $i = 1, 2, \dots, N$; m – кількість компонентів вектору $\mathbf{Z}$. В нашому випадку m дорівнює 3.

Вектор $\mathbf{Z}$ отримують шляхом перетворення негаусівського випадкового вектору $\mathbf{X} = \{X_1, X_2, \dots, X_m\}^T$ за допомогою багатовимірного нормалізуючого перетворення

$$\mathbf{Z} = \psi(\mathbf{X}). \quad (1.5)$$

В нашому випадку у якості перетворення (1.5) ми використовуємо перетворення у формі десяткового логарифму.

Відомо [11], якщо дані мають багатовимірний нормальний розподіл, то розподіл квадрату MD поводитьсь як розподіл χ^2 . Для великих $N - m$ (принаймні більше 25), відстань d_i^2 повинна вести себе приблизно як незалежні випадкові величини $\chi_{m,\alpha}^2$. Тут $\chi_{m,\alpha}^2$ – це квантиль розподілу χ^2 з рівнем значимості α . Значення даних, для яких величина квадрату MD більша, ніж квантиль розподілу χ^2 , розглядаються як викиди, і ці значення відкидаються. Після відкидання зменшене число багатомірних точок даних знову нормалізується, використовуючи перетворення (1.5), доки всі значення квадрату MD (1.3) будуть меншими чи дорівнювати квантилю розподілу χ^2 .

Для нормалізованих за десятковим логарифмом даних (з перетворенням) з табл. 1.1 є тривимірний викид у рядку 18 тому, що значення квадрату відстані Махаланобіса для значень метрик з цього рядка більше за значення квантилю розподілу χ^2 , який дорівнює 12,84 для 3 ступенів свободи та рівня значимості 0,005. Зверніть увагу, що для даних метрик з табл.1.1 без нормалізації або перетворення є три рядки даних (рядки 2, 18 та 35), для яких значення квадрату відстані Махаланобіса більше за значення квантилю розподілу χ^2 12,84 для 3 ступенів свободи та рівня значимості 0,005. Це пояснюється тим, що тривимірний розподіл даних метрик з табл.1.1 не є нормальним.

Тому ми відкидаємо рядок 18 та знову перевіряємо але вже 48 строк даних метрик з табл.1.1 без рядка 18 на наявність викидів. Для нормалізованих за десятковим логарифмом 48 строк даних з табл. 1.1 (без рядка 18) є тривимірний викид у рядку 30 тому, що значення квадрату відстані Махаланобіса для значень метрик з цього рядка, більше за значення квантилю

розподілу χ^2 , який дорівнює 12,84 для 3 ступенів свободи та рівня значимості 0,005.

Тому ми відкидаємо рядок 30 та знову перевіряємо але вже 47 строк даних метрик з табл.1.1 (без рядків 18 та 30) на наявність викидів. Для нормалізованих за десятковим логарифмом 47 строк даних з табл. 1.1 (без рядків 18 та 30) викиди відсутні тому, що всі значення квадрату відстані Махаланобіса у цьому випадку менші за значення квантилю розподілу χ^2 , який дорівнює 12,84 для 3 ступенів свободи та рівня значимості 0,005. При цьому за тестом Мардіа багатовимірний розподіл нормалізованих даних є гаусівським.

За даними метрик з 47 рядків з табл. 1.1 (без рядків 18 та 30) була побудована лінійна регресійна модель (1.2), для якої за методом найменших квадратів оцінки параметрів $\hat{b}_0$, $\hat{b}_1$ та $\hat{b}_2$ мають відповідно такі значення: -1,4300, 1,1331 та 0,27018. Оцінка дисперсії $\hat{\sigma}_\varepsilon^2$ дорівнює 8,190. При цьому розподіл випадкової величини ε є нормальним. Показники точності відповідного оцінювання метрики RFC за допомогою лінійної регресійної моделі (1.2) у цьому випадку є такими: значення MMRE дорівнює 0,1432, а значення PRED(0,25) складає 0,8511.

Середня величина відносної похибки MMRE та відсоток прогнозування на рівні величини відносної помилки 0,25, PRED(0,25) зазвичай застосовуються для оцінювання якості прогнозування за допомогою регресійних моделей і в інженерії програмного забезпечення [14, 15]. Допустимі значення MMRE і PRED(0,25) складають не більше 0,25 і не менше 0,75 відповідно.

Значення величини відносної похибки MRE обчислюється як

$$MRE_i = \left| \frac{Y_i - \hat{Y}_i}{Y_i} \right|.$$

Середня величина відносної похибки $MMRE$ визначається як

$$MMRE = \frac{1}{N} \sum_{i=1}^N MRE_i. \quad (1.6)$$

Зазвичай $MMRE \leq 0,25$ вважається прийнятною точністю моделі.

Значення $PRED(0,25)$ обчислюється як

$$PRED(0,25) = \frac{1}{N} \sum_{i=1}^N \begin{cases} 1 & \text{if } MRE_i \leq 0,25 \\ 0 & \text{otherwise.} \end{cases} \quad (1.7)$$

Зазвичай $PRED(0,25) \geq 0,75$ вважається прийнятною точністю прогнозування за допомогою регресійних моделей.

Хоча отримані значення $MMRE$ та $PRED(0,25)$ для лінійної регресійної моделі (1.2) є достатньо непоганими, але вони можуть бути покращені за рахунок використання нелінійної регресійної моделі.

Таким чином, виникає необхідність у проведенні досліджень за обраною темою.

1.3 Висновки за розділом 1

У даному розділі наведено огляд існуючих математичних моделей для оцінювання метрики RFC застосунків на Java, здійснено обґрунтування необхідності проведення досліджень за обраною темою.

1) Для оцінювання метрики RFC застосунків на Java існують лише лінійні регресійні моделі, які залежать від двох метрик: СВО та WMC. Але лінійні моделі не завжди мають кращі показники точності у порівнянні з нелінійними регресійними моделями. Тому виникає потреба в побудові

відповідних нелінійних моделей для оцінювання метрики RFC застосунків з відкритим кодом на Java.

2) Побудову нелінійної регресійної моделі для оцінювання метрики RFC застосунків з відкритим кодом на Java, краще здійснювати за методом, що заснований на нормалізуючих перетвореннях та інтервалах прогнозування. Тому удосконалювати регресійну модель для оцінювання метрики RFC застосунків з відкритим кодом на Java, доцільно саме на основі зазначеного методу.

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

2.1 Перевірка даних для побудови нелінійної регресійної моделі для оцінювання метрики RFC застосунків з відкритим кодом на Java, на наявність викидів

Для побудови нелінійної регресійної моделі для оцінювання метрики RFC застосунків з відкритим кодом на Java, використовується вибірка тривимірних негаусових даних з табл.1.1 за метриками RFC, CBO та WMC, які позначені відповідно як Y , X_1 та X_2 .

Для виявлення викидів у цих даних з табл.1.1 ми використовували метод, оснований на нормалізуючих перетвореннях і квадраті відстані Махаланобіса. Значення квадрату MD для кожної багатовимірної точки нормалізованих даних i , $i = 1, 2, \dots, N$, ми визначали за (1.3).

За даними табл.1.1 для нормалізованих за десятковим логарифмом даних (з перетворенням) є два тривимірних викиди для рівня значимості 0,005 – це рядки даних 18 та 30. При цьому за тестом Мардіа багатовимірний розподіл нормалізованих даних є гаусівським.

В подальшому 47 рядків тривимірних негаусових даних (без рядків 18 та 30), що наведені у табл.1.1, використовуються для побудови нелінійної регресійної моделі для оцінювання метрики RFC застосунків з відкритим кодом на Java.

2.2 Побудова нелінійної регресійної моделі для оцінювання метрики RFC застосунків з відкритим кодом на Java

Для побудови нелінійної регресійної моделі для оцінювання метрики RFC застосунків з відкритим кодом на Java, ми використовуємо метод, що заснований на нормалізуючих перетвореннях та інтервалах прогнозування [7, 8], а також модифікований метод [9], який ще додатково враховує можливу наявність викидів у нелінійній регресійній моделі [16]. В загальному випадку процес побудови нелінійної регресійної моделі за цим методом є ітераційним. Як ми описували раніше, у даних з табл.1.1 за дві ітерації були визначені два тривимірні викиди: рядків 18 та 30. Далі за цим методом вже на третій ітерації спочатку тривимірні негаусівськи дані з табл.1.1 (без рядків 18 та 30) ми знову нормалізуємо за допомогою нормалізуючого перетворення у формі десяткового логарифму. При цьому тривимірних викидів у даних не було. Про це свідчать значення квадрату відстані Махаланобіса на третій ітерації, які позначені в табл.2.1 як SMD_3 . Викиди відсутні тому, що всі значення квадрату відстані Махаланобіса у цьому випадку менші за значення квантилю розподілу χ^2 , який дорівнює 12,84 для 3 ступенів свободи та рівня значимості 0,005.

Далі для нормалізованих даних з 47 рядків будемо лінійну регресійну модель для визначення нормалізованого значення метрики RFC

$$Z_Y = \hat{Z}_Y + \varepsilon = \hat{b}_0 + \hat{b}_1 Z_1 + \hat{b}_2 Z_2 + \hat{b}_3 Z_3 + \varepsilon, \quad (2.1)$$

де оцінки параметрів $\hat{b}_0$, $\hat{b}_1$ та $\hat{b}_2$ мають відповідно такі значення: 0,00511, 0,59705 та 0,46092, а оцінка середньо квадратичного відхилення $\hat{\sigma}_\epsilon$ випадкової величини ϵ дорівнює 0,0709. Параметри моделі (2.1) оцінювалися за методом найменших квадратів.

Після цього будуюмо нелінійну регресійну модель за лінійною моделлю (2.1) та перетворенням у вигляді десяткового логарифму

$$Y = 10^{\epsilon + \hat{b}_0} X_1^{\hat{b}_1} X_2^{\hat{b}_2} X_3^{\hat{b}_3}. \quad (2.2)$$

Далі згідно з [7], на третьому етапі ми визначаємо границі інтервалу прогнозування нелінійної регресії для рівня значущості, що дорівнює 0,05. Згідно з [6], границі інтервалу прогнозування нелінійної регресії визначаються як

$$\psi_Y^{-1} \left(\hat{Z}_Y \pm t_{\alpha/2, v} S_{Z_Y} \left\{ 1 + \frac{1}{N} + (\mathbf{z}_X^+)^T \left[(\mathbf{Z}_X^+)^T \mathbf{Z}_X^+ \right]^{-1} (\mathbf{z}_X^+) \right\}^{1/2} \right), \quad (2.3)$$

де ψ_Y^{-1} – зворотне до нормалізуючого перетворення ψ_Y для залежної змінної Y ; $\hat{Z}_Y$ – результат прогнозування лінійного регресійного рівняння для значень компонент вектору $\mathbf{z}_X = \{Z_1, Z_2, \dots, Z_k\}$, що отриманий за нормалізованими даними; k – кількість незалежних змінних (у нашому випадку $k=2$); $\mathbf{Z}_X^+$ – матриця центрованих нормалізованих незалежних змінних, яка містить значення $Z_{1i} - \bar{Z}_1$, $Z_{2i} - \bar{Z}_2$, ..., $Z_{ki} - \bar{Z}_k$; $t_{\alpha/2, v}$ – квантиль t -розподілу Стюдента в

залежності від рівня значимості $\alpha/2$ та кількості ступенів свободи v ;

$$S_{Z_Y}^2 = \frac{1}{v} \sum_{i=1}^N (Z_{Y_i} - \hat{Z}_{Y_i})^2, \quad v = N - k - 1; \quad (\mathbf{Z}_X^+)^T \mathbf{Z}_X^+ - k \times k \text{ матриця}$$

$$(\mathbf{Z}_X^+)^T \mathbf{Z}_X^+ = \begin{pmatrix} S_{Z_1 Z_1} & S_{Z_1 Z_2} & \dots & S_{Z_1 Z_k} \\ S_{Z_1 Z_2} & S_{Z_2 Z_2} & \dots & S_{Z_2 Z_k} \\ \dots & \dots & \dots & \dots \\ S_{Z_1 Z_k} & S_{Z_2 Z_k} & \dots & S_{Z_k Z_k} \end{pmatrix}, \quad (2.4)$$

$$\text{де } S_{Z_q Z_r} = \sum_{i=1}^N [Z_{q_i} - \bar{Z}_q][Z_{r_i} - \bar{Z}_r], \quad q, r = 1, 2, \dots, k.$$

В (2.3) знаки мінус та плюс відповідають нижній та верхній границям інтервалу прогнозування нелінійної регресії, що побудована на основі нормалізуючого перетворення.

На третій ітерації матриця (2.4) для нормалізованих за перетворенням у формі десяткового логарифму 47 рядків даних з табл.1.1 (без рядків 18 та 30) є такою

$$(\mathbf{Z}_X^+)^T \mathbf{Z}_X^+ = \begin{pmatrix} 0,93285 & 0,64299 \\ 0,64299 & 1,88517 \end{pmatrix}.$$

Границі інтервалу прогнозування нелінійної регресії, що отримані за допомогою перетворення у формі десяткового логарифму на третій ітерації, наведені у табл.2.1. В табл.2.1 ліва границя інтервалу прогнозування на третій ітерації позначена як LB₃, а права – як UB₃. Для моделі (2.2) з оцінками параметрів, що отримані для 47 рядків даних з табл.1.1 (без рядків 18 та 30),

виявилося, що значення Y для одного застосунку (рядок 44) виходить за визначені межі інтервалу прогнозування.

Таблиця 2.1 – Границі інтервалу прогнозування нелінійної регресії, що отримані за допомогою перетворення у формі десяткового логарифму на двох ітераціях

№	Y	LB_3	UB_3	SMD_3	LB_4	UB_4	SMD_4
1	2	3	4	5	6	7	8
1	9,864	8,47	16,91	3,08	8,64	16,36	3,18
2	26,448	21,82	46,46	12,60	21,69	43,64	12,31
3	8,688	8,63	17,16	5,49	8,66	16,53	5,91
4	15,389	11,40	22,49	0,07	11,55	21,65	0,06
5	23,885	18,70	37,43	2,36	19,48	36,96	2,55
6	10,488	8,21	16,33	1,49	8,27	15,61	1,44
7	31,526	19,85	39,80	2,79	20,56	39,09	2,74
8	11,143	6,38	13,23	8,58	6,12	12,06	11,06
9	37,278	23,59	48,03	4,91	24,88	48,03	4,80
10	7,306	6,15	12,68	7,65	5,92	11,60	7,74
11	21,130	12,13	24,04	2,27	12,50	23,50	2,48
12	32,010	17,57	35,24	4,71	18,43	35,06	4,71
13	14,556	11,09	21,89	0,21	11,23	21,05	0,18
14	13,428	9,63	19,20	1,43	9,90	18,72	1,38
15	10,868	7,88	15,71	1,38	7,93	15,00	1,40
16	12,488	11,13	21,98	1,97	11,28	21,14	2,07
17	19,467	10,36	20,75	5,16	10,74	20,40	5,42
18	43,225	-	-	-	-	-	-
19	22,947	13,88	27,79	2,94	13,87	26,36	3,61
20	9,077	8,04	16,00	3,10	8,08	15,26	3,08
21	16,351	10,16	20,63	4,58	9,93	19,14	5,76
22	22,215	13,33	26,43	1,71	13,78	25,91	1,78

Продовження табл. 2.1

1	2	3	4	5	6	7	8
23	12,969	10,44	20,64	0,84	10,49	19,70	0,80
24	15,020	9,94	19,65	0,44	10,05	18,86	0,61
25	19,751	15,84	31,85	3,27	16,67	31,79	3,75
26	14,125	10,83	21,64	2,05	10,71	20,34	2,12
27	24,214	16,61	33,03	1,12	17,19	32,43	1,08
28	14,922	12,15	23,98	0,70	12,37	23,17	0,72
29	12,653	10,44	20,63	1,06	10,61	19,91	1,08
30	18,124	-	-	-	-	-	-
31	9,521	7,57	15,31	4,10	7,76	14,87	4,09
32	25,931	16,58	32,98	1,50	17,17	32,39	1,47
33	16,265	9,63	19,17	2,38	9,87	18,63	2,58
34	18,629	10,14	20,04	2,93	10,23	19,19	3,89
35	37,158	26,97	56,14	7,90	27,67	54,45	7,72
36	12,974	10,74	21,34	1,89	10,69	20,18	1,84
37	12,596	9,98	19,73	0,75	10,12	19,01	0,73
38	24,936	18,67	37,33	1,97	19,39	36,77	2,03
39	15,586	9,83	19,45	0,87	9,88	18,55	1,29
40	25,388	19,12	38,67	3,66	19,39	37,15	3,56
41	18,966	16,35	32,52	2,60	16,96	32,01	3,03
42	16,735	9,76	19,32	1,84	9,81	18,43	2,57
43	20,526	13,61	27,01	0,97	14,09	26,54	0,93
44	16,370	7,49	15,23	11,39	-	-	-
45	10,622	7,77	15,53	1,78	7,69	14,60	1,97
46	13,100	9,81	19,42	0,47	9,96	18,70	0,44
47	15,241	9,96	19,93	2,13	10,30	19,53	2,08
48	15,329	10,03	20,06	2,09	10,37	19,66	2,04
49	22,707	16,61	33,03	1,79	18,06	34,19	1,92

Згідно з [11, 12], ми відкинули рядок 44 як викид та перейшли до четвертої ітерації побудови нелінійної регресійної моделі (2.2).

На четвертій ітерації, згідно з [6, 7], ми знову нормалізували дані з 46 рядків з табл.1.1 (за винятком відкинутих рядків 18, 30 та 44) за допомогою нормалізуючого перетворення у формі десяткового логарифму. При цьому тривимірних викидів у даних не було. Про це свідчать значення квадрату відстані Махаланобіса на четвертій ітерації, які позначені в табл.2.1 як SMD_4 . Викиди відсутні тому, що всі значення квадрату відстані Махаланобіса у цьому випадку менші за значення квантилю розподілу χ^2 , який дорівнює 12,84 для 3 ступенів свободи та рівня значимості 0,005.

Границі інтервалу прогнозування нелінійної регресії, що отримані за допомогою перетворення у формі десяткового логарифму на четвертій ітерації, наведені у табл.2.1. В табл.2.1 ліва границя інтервалу прогнозування на четвертій ітерації позначена як LB_4 , а права – як UB_4 .

Матриця (2.2) для нормалізованих за перетворенням у формі десяткового логарифму 46 рядків даних з табл.1.1 (за винятком відкинутих рядків 18, 30 та 44) є такою

$$(\mathbf{Z}_x^+)^T \mathbf{Z}_x^+ = \begin{pmatrix} 0,85544 & 0,62713 \\ 0,62713 & 1,88192 \end{pmatrix}.$$

Границі інтервалу прогнозування на четвертій ітерації з табл.2.1 свідчать про те, що викиди у регресії відсутні. Тому на цьому побудову нелінійної регресійної моделі (2.2) завершено.

Остаточна нелінійна регресійна модель (2.2) має наступні оцінки параметрів $\hat{b}_0$, $\hat{b}_1$ та $\hat{b}_2$: -0,05477, 0,67226 та 0,44154, а оцінка середнього квадратичного відхилення $\hat{\sigma}_\varepsilon$ випадкової величини ε дорівнює 0,0654.

Побудована нелінійна регресійна модель (2.2) 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). Значення R^2 , MMRE і PRED(0,25) для створеної моделі дорівнюють відповідно 0,8718, 0,1262 та 0,9783. Це свідчить про задовільну якість отриманої моделі. Зазначимо, що показники MMRE і PRED(0,25) обчислювалися за формулами (1.6) та (1.7) відповідно.

Вибірковий коефіцієнт детермінації R^2 визначався як

$$R^2 = 1 - \frac{SS_E}{SS_T},$$

де SS_E – сума квадратів залишків (the residual sum of squares) або сума квадратів помилок (the error sum of squares), $SS_E = \sum_{i=1}^N (Y_i - \hat{Y}_i)^2$; SS_T – загальна сума

квадратів (the total sum of squares), $SS_T = \sum_{i=1}^N (Y_i - \bar{Y})^2$. Відомо, що

$SS_T = SS_R + SS_E$. Тут SS_R – сума квадратів регресії (the regression sum of squares),

$$SS_R = \sum_{i=1}^N (\hat{Y}_i - \bar{Y})^2.$$

Коефіцієнт детермінації R^2 інтерпретується як частка мінливості у спостережуваній залежній змінній Y , що пояснюється моделлю лінійної регресії. Значення R^2 вказує наскільки емпіричні дані підтверджують математичну модель або, чи краща наша модель за вибіркове середнє. З формули для R^2 , $0 \leq R^2 \leq 1$, значення R^2 вважається прийнятним, якщо воно більше за 0,75. Велике значення R^2 свідчить про те, що модель успішно пояснює мінливість Y .

Коли значення R^2 є малим, то це може свідчити про те, що потрібно знайти альтернативну модель, яка може врахувати більшу мінливість Y .

Показники точності нелінійної регресійної моделі (2.2) при оцінюванні метрики RFC застосунків з відкритим кодом на Java кращі за значення R^2 , MMRE та PRED(0,25) для лінійної регресійної моделі (1.2), які дорівнюють відповідно 0,8629, 0,1327 та 0,8696. Таким чином, нелінійна регресійна модель (2.2) має кращий показник PRED(0,25) у порівнянні з лінійною регресійною моделлю (1.2) на 12,5%.

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 [17] 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 [9].

2.3 Висновки за розділом 2

У даному розділі виконано перевірку даних для побудови нелінійної регресійної моделі для оцінювання метрики RFC застосунків з відкритим кодом на Java, на наявність тривимірних викидів, побудовано нелінійну регресійну модель для оцінювання метрики RFC застосунків з відкритим кодом на Java.

1) Перевірку даних для побудови нелінійної регресійної моделі для оцінювання метрики RFC застосунків з відкритим кодом на Java на наявність викидів виконано за методом, який оснований на нормалізуючих перетвореннях і квадраті відстані Махаланобіса. З'ясовано, що за методом, який оснований на нормалізуючих перетвореннях і квадраті відстані Махаланобіса, тривимірні дані містять два багатовимірних викиди. Застосування критерія на основі квадрата відстані Махаланобіса до даних без їх нормалізації показує, що дані містять три багатовимірних викиди, які не можуть бути всі визнані викидами завдяки тому, що розподіл цих даних не є нормальним.

2) Удосконалено двофакторну нелінійну регресійну модель для оцінювання метрики RFC в залежності від метрик CBO та WMC застосунків з відкритим кодом на Java на основі нормалізуючого перетворення у формі десяткового логарифму із врахуванням викидів у регресії. Ця нелінійна регресійна модель у порівнянні з існуючими регресійними моделями, що побудовані без врахуванням викидів у регресії, має більші значення множинного коефіцієнту детермінації і відсотка прогнозування, менші значення середньої величини відносної похибки та ширин довірчого інтервалу і інтервалу прогнозування для більшої кількості значень метрик CBO та WMC. Зазначена нелінійна регресійна модель у порівнянні з лінійною має кращий показник $PRED(0,25)$ на 12,5%.

3) Отримані результати свідчать про те, що для побудови нелінійних регресійних моделей для оцінювання метрики RFC застосунків з відкритим кодом на Java, потрібно використовувати нормалізуючі перетворення з урахуванням викидів у нелінійній регресії.

РОЗРОБКА ПРОЕКТУ ПРОГРАМИ ДЛЯ ОЦІНЮВАННЯ МЕТРИКИ RFC ЗАСТОСУНКІВ З ВІДКРИТИМ КОДОМ НА JAVA

3.1 Постановка задачі на розробку проекту програми для оцінювання метрики RFC застосунків з відкритим кодом на Java

Виходячи з аналізу існуючих моделей для оцінювання метрики RFC застосунків з відкритим кодом на Java, сформулюємо постановку задачі. Потрібно створити програму для оцінювання метрики RFC застосунків з відкритим кодом на Java, яка би задовольняла таким вимогам.

Вимоги до програми.

Вимоги функціональних характеристик.

Вимоги до складу функцій, що виконуються.

Програма має виконувати такі функції:

- 1) Вводити з текстового файлу емпіричні дані з метрик RFC, CBO та WMC застосунків з відкритим кодом на Java.
- 2) Вводити з інтерфейсу програми довірчу ймовірність, дві метрики CBO та WMC застосунку, для якого потрібно здійснити оцінювання метрики RFC.
- 3) Виконувати нормалізацію емпіричних даних за нормалізуючим перетворенням у формі десяткового логарифму.
- 4) Знаходити значення коефіцієнтів лінійного рівняння регресії для нормалізованих емпіричних даних.
- 5) Здійснювати побудову нелінійної регресійної моделі.

6) Обчислювати значення коефіцієнту детермінації R^2 та величини відносної похибки MRE.

7) Здійснювати побудову інтервалу прогнозування та довірчого інтервалу нелінійної регресії.

8) Виводити на екран нелінійну регресію, довірчий інтервал та інтервал прогнозування нелінійної регресії метрики RFC, значення коефіцієнту детермінації R^2 та величини відносної похибки MRE.

9) Виводити у текстовий файл результати оцінювання метрики RFC застосунку з відкритим кодом на Java.

Вимоги до складу й параметрів технічних засобів.

Програма має експлуатуватись на обладнанні з параметрами не нижчими за наступні: процесор Pentium IV (або еквівалентний) з інструкціями SSE2, оперативна пам'ять 2 Гб (1 Гб мінімум), простір на жорсткому диску 600 Мб мінімум.

Вимоги до інформаційної та програмної сумісності.

Для запуску та роботи з програмою необхідна операційна система Windows Vista/7/8/10 (32 або 64 біти), система моделювання Scilab 6.0.0., компілятор C (Visual Studio 2010 або Visual Express 2010) для компіляції зовнішніх модулів C або C++ та використання моделі в Xcos.

На підставі цієї постановки задачі розроблено технічне завдання, яке наведено у додатку А.

3.2 Розробка ескізного проекту програми для оцінювання метрики RFC застосунків з відкритим кодом на Java

В ескізному проекті було виявлено варіанти використання та побудовано їх сценарії, розроблено зовнішню специфікацію програми для оцінювання метрики RFC застосунків з відкритим кодом на Java, та стани цієї програми при виконанні.

Даний проект передбачає наявність лише одного актора – користувача програми, який буде повноцінно використовувати всі функції програми. Виявлені варіанти використання знаходяться в таблиці 3.1.

Таблиця 3.1 Виявлені варіанти використання

Основні актори	Найменування	Формулювання
1	2	3
Користувач	Ввести дані метрик застосунків	Вводить емпіричні дані з метрик RFC, CBO та WMC застосунків з відкритим кодом на Java
Користувач	Ввести метрики CBO та WMC застосунку	Вводить метрики CBO та WMC застосунку, за якими потрібно оцінити метрику RFC
Користувач	Ввести довірчу ймовірність	Вводить довірчу ймовірність, для якої потрібно оцінити довірчий інтервал та інтервал прогнозування нелінійної регресії метрики RFC застосунку з відкритим кодом на Java
Користувач	Нормалізувати дані	Виконує нормалізацію введених даних

Продовження табл. 3.1

1	2	3
Користувач	Побудувати нелінійну регресійну модель	Будує нелінійну регресійну модель для оцінювання значення метрики RFC застосунків з відкритим кодом на Java
Користувач	Отримати довірчий інтервал та інтервал прогнозування	Отримує із заданою ймовірністю довірчий інтервал та інтервал прогнозування нелінійної регресії для інтервального оцінювання метрики RFC

Діаграма варіантів використання програми для оцінювання метрики RFC застосунків з відкритим кодом на Java, зображена на рисунку 3.1.

Сценарії варіантів використання зображені у вигляді діаграми діяльності на рисунку 3.2.

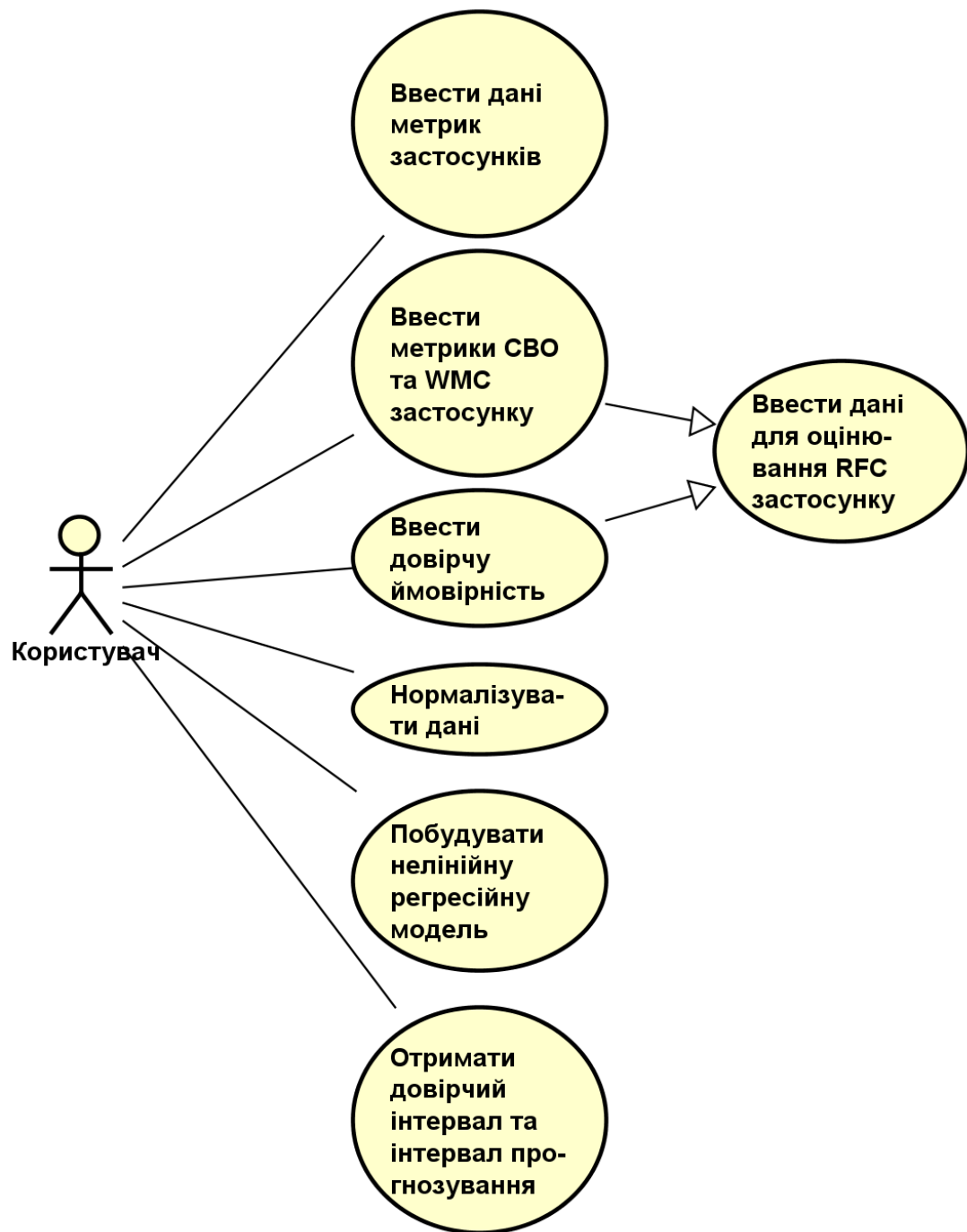


Рисунок 3.1 – Діаграма варіантів використання програми для оцінювання метрики RFC застосунків з відкритим кодом на Java

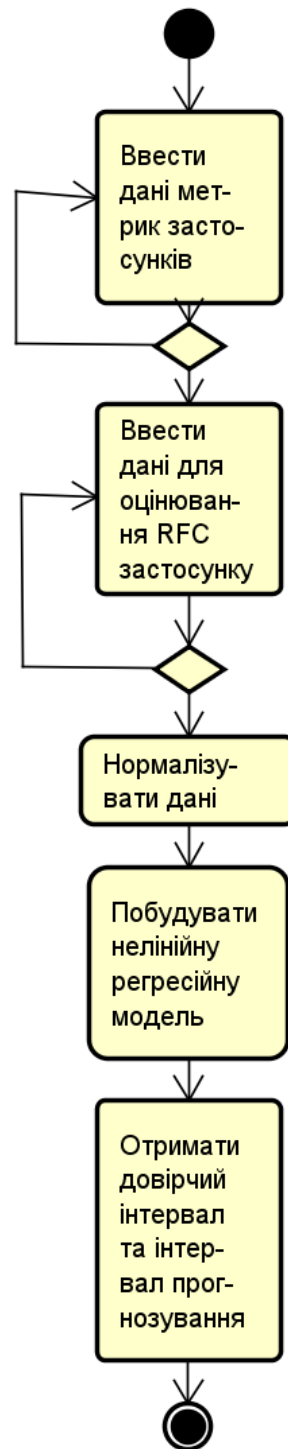


Рисунок 3.2 – Діаграма діяльності ПЗ для оцінювання метрики RFC застосунків з відкритим кодом на Java

Зовнішня специфікація програми.

Назва програми: RFC_Estim_LG.

Функції, які виконує програма:

1) Вводити з текстового файлу емпіричні дані з метрик RFC, CBO та WMC застосунків з відкритим кодом на Java.

2) Вводити з інтерфейсу програми довірчу ймовірність, дві метрики CBO та WMC застосунку, для якого потрібно здійснити оцінювання метрики RFC.

3) Виконувати нормалізацію емпіричних даних за нормалізуючим перетворенням у формі десяткового логарифму.

4) Знаходити значення коефіцієнтів лінійного рівняння регресії для нормалізованих емпіричних даних.

5) Здійснювати побудову нелінійної регресійної моделі.

6) Обчислювати значення коефіцієнту детермінації R^2 та величини відносної похибки MRE.

7) Здійснювати побудову інтервалу прогнозування та довірчого інтервалу нелінійної регресії.

8) Виводити на екран нелінійну регресію, довірчий інтервал та інтервал прогнозування нелінійної регресії метрики RFC, значення коефіцієнту детермінації R^2 та величини відносної похибки MRE.

9) Виводити у текстовий файл результати оцінювання метрики RFC застосунку з відкритим кодом на Java.

Вхідні дані:

- емпіричні дані з метрик RFC, CBO та WMC застосунків з відкритим кодом на Java;

- значення довірчої ймовірності, метрик CBO та WMC застосунку, для якого потрібно здійснити оцінювання метрики RFC.

Вихідні дані:

- значення нелінійної регресії, границь довірчого інтервалу та інтервалу прогнозування нелінійної регресії метрики RFC застосунку.

- значення коефіцієнту детермінації та відносної похибки MRE.

Зовнішні ефекти: поява графічного вікна для вводу даних, управління виводом даних, графічного вікна для виводу даних, нелінійної регресії, довірчого інтервалу та інтервалу прогнозування нелінійної регресії метрики RFC застосунку, вихід з програми (закриття всіх графічних вікон).

Для проектування інтерфейсу програми для оцінювання метрики RFC застосунків з відкритим кодом на Java представимо стани програми при виконанні у вигляді відповідної діаграми станів. Діаграма станів програми для оцінювання метрики RFC застосунків з відкритим кодом на Java, наведена на рисунку 3.3.

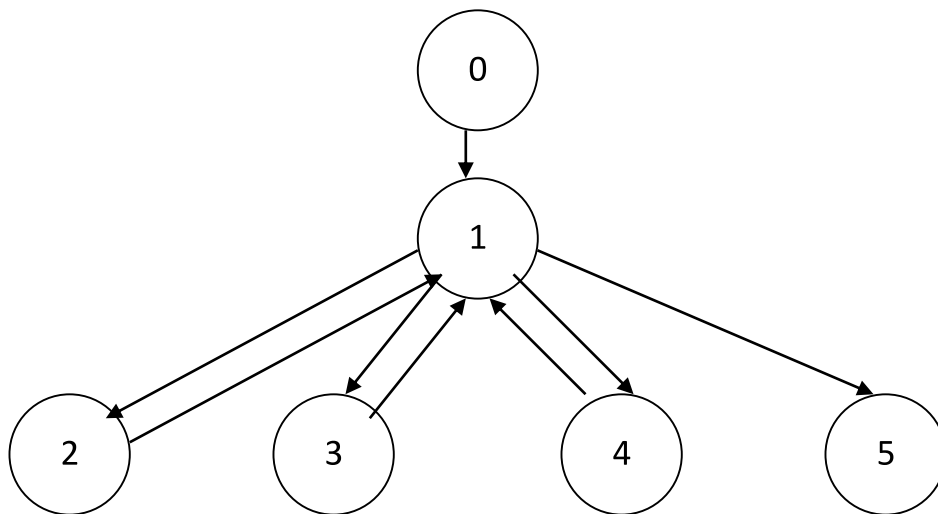


Рисунок 3.3 – Діаграма станів програми для оцінювання метрики RFC застосунків з відкритим кодом на Java

На діаграмі станів програми введені такі позначення станів: 0 – початковий стан, 1 – графічне вікно головного модуля, 2 – графічне вікно візуалізації введених емпіричних даних, 3 – графічне вікно із результатами знаходження нелінійної регресії, 4 – графічне вікно з результатами побудови довірчого інтервалу та інтервал прогнозування нелінійної регресії метрики RFC застосунку з відкритим кодом на Java, 5 – кінцевий стан.

Далі виконуємо розробку технічного проекту програми для оцінювання метрики RFC застосунків з відкритим кодом на Java

3.3 Розробка технічного проекту програми для оцінювання метрики RFC застосунків з відкритим кодом на Java

В технічному проекті було спроектовано програму для оцінювання метрики RFC застосунків з відкритим кодом на Java на основі покрокової деталізації зверху вниз, розроблено специфікації модулів програми, побудовано алгоритми модулів програми.

Програма складається з таких модулів: головного модуля (RFC_Estim_LG), модуля для введення емпіричних даних з файлу та їх візуалізації (DataInput), модуля для побудови нелінійної регресії, довірчого інтервалу та інтервалу прогнозування нелінійної регресії метрики RFC застосунків з відкритим кодом на Java (RFC_AppEstim), модуля для закриття графічних вікон (W_Close).

Специфікація головного модуля.

Назва модуля: RFC_Estim_LG.

Функції, які виконує програма:

- 1) Відкриття графічного вікна головного модуля.
- 2) Введення імені файлу з емпіричними даними.
- 3) Виклик функції для введення емпіричних даних з файлу та їх візуалізації у графічному вікні.
- 4) Вибір наявності фактичного метрики RFC застосунку з відкритим кодом на Java.
- 5) Виклик функції для визначення нелінійної регресії, довірчого інтервалу та інтервалу прогнозування нелінійної регресії метрики RFC застосунків з відкритим кодом на Java.
- 6) Виклик функції для закриття графічних вікон.

Вхідні дані:

- емпіричні дані з метрик RFC, CBO та WMC застосунків з відкритим кодом на Java;
- значення довірчої ймовірності, метрик CBO та WMC застосунку, для якого потрібно здійснити оцінювання метрики RFC.

Вихідні дані:

- значення нелінійної регресії, границь довірчого інтервалу та інтервалу прогнозування нелінійної регресії метрики RFC застосунку з відкритим кодом на Java.
- значення коефіцієнту детермінації.

Зовнішні ефекти: поява графічного вікна для вводу даних, управління виводом даних, побудови довірчого інтервалу та інтервалу прогнозування нелінійної регресії метрики RFC застосунку з відкритим кодом на Java, вихід з програми (закриття всіх графічних вікон).

Специфікація модуля для введення емпіричних даних з файлу та їх візуалізації.

Назва модуля: DataInput.

Функції, які виконує модуль:

- 1) Введення емпіричних даних (значень випадкових величини) з файлу.
- 2) Формування одномірних та двомірних масивів значень випадкових величини (емпіричних даних).
- 3) Відкриття графічного вікна для візуалізації введених емпіричних даних.
- 4) Візуалізацію введених емпіричних даних.

Вхідні дані:

- значення випадкових величини (емпіричні дані).

Вихідні дані:

- одномірні та двомірні масиви значення випадкових величини (емпіричні дані);
- кількість значень випадкових величин (емпіричних даних);
- ідентифікатори графічних вікон.

Зовнішні ефекти: Поява графічного вікна для візуалізації введених емпіричних даних.

Специфікація модуля для визначення нелінійної регресії, довірчого інтервалу та інтервалу прогнозування нелінійної регресії метрики RFC застосунків з відкритим кодом на Java.

Назва модуля: RFC_AppEstim.

Функції, які виконує модуль:

1) Визначає нелінійну регресію, довірчий інтервал та інтервалу прогнозування нелінійної регресії метрики RFC застосунків з відкритим кодом на Java.

2) Обчислює значення коефіцієнту детермінації для нелінійної регресії та величини відносної похибки MRE.

Вхідні дані:

- двомірні масиви значень випадкових величини (емпіричних даних);
- кількість значень випадкових величин (емпіричних даних).

Вихідні дані:

- значення нелінійної регресії, нижньої, верхньої границь довірчого інтервалу та інтервалу прогнозування нелінійної регресії метрики RFC застосунків з відкритим кодом на Java;

- значення коефіцієнту детермінації для нелінійної регресії та величини відносної похибки MRE.

Зовнішні ефекти: з'являються результати нелінійної регресії, нижньої та верхньої границь довірчого інтервалу та інтервалу прогнозування нелінійної регресії метрики RFC застосунків з відкритим кодом на Java та виводиться значення коефіцієнту детермінації для нелінійної регресії та величини відносної похибки MRE.

Назва модуля: W_Close.

Функції, які виконує модуль:

Закриття графічних вікон.

Вхідні дані:

- ідентифікатори графічних вікон.

Вихідні дані: відсутні.

Зовнішні ефекти: Закриття всіх графічних вікон.

В подальшому були розроблені алгоритми зазначених модулів.

3.4 Розробка робочого проекту програми для оцінювання метрики RFC застосунків з відкритим кодом на Java

В робочому проекті було розроблено програму для оцінювання метрики RFC застосунків з відкритим кодом на Java здійснено тестування програми, розроблено програмну документацію, здійснено випробування програми.

Спочатку scі-мовою системи моделювання Scilab було виконано кодування алгоритмів, що були розроблені на стадії технічного проекту. Текст програми наведений в додатку Б. Приклад кодування модуля RFC_AppEstim, що здійснює оцінювання метрики RFC застосунків з відкритим кодом на Java, scі-мовою в системі моделювання Scilab наведено нижче.

```
//Функція оцінювання метрики RFC застосунку з відкритим кодом на
Java
function RFC_AppEstim(x, yx, n)
//Считываем значение переменных из текстовых полей и
//преобразовываем их к числовому типу.
x1i=eval(get(edit_text4, 'string'));
x2i=eval(get(edit_text5, 'string'));
//x3i=eval(get(edit_text6, 'string'));
Pconf=eval(get(edit_text6, 'string'));

x1A=0; x2A=0;
yA=0;
for i=1:n
    x1A=x1A+x(i,1); x2A=x2A+x(i,2);
    yA=yA+yx(i,1);
end;
x1A=x1A/n; x2A=x2A/n;
yA=yA/n;
Sx1x1=0; Sx2x2=0;
Sx1x2=0;
```

```

SyA=0;
for i=1:n
    Sx1x1=Sx1x1+(x(i,1)-x1A)*(x(i,1)-x1A);
    Sx1x2=Sx1x2+(x(i,1)-x1A)*(x(i,2)-x2A);
    Sx2x2=Sx2x2+(x(i,2)-x2A)*(x(i,2)-x2A);
    SyA=SyA+(y(x(i,1))-yA)*(y(x(i,1))-yA);
end;
Sxx(1,1)=Sx1x1; Sxx(1,2)=Sx1x2;
Sxx(2,1)=Sx1x2; Sxx(2,2)=Sx2x2;

Sxx_1=1/Sxx;

z1A=0; z2A=0;
for i=1:n
    z1A=z1A+log10(x(i,1)); z2A=z2A+log10(x(i,2));
end;
z1A=z1A/n; z2A=z2A/n;
Sz1z1=0; Sz2z2=0;
Sz1z2=0;
for i=1:n
    Sz1z1=Sz1z1+(log10(x(i,1))-z1A)*(log10(x(i,1))-z1A);
    Sz1z2=Sz1z2+(log10(x(i,1))-z1A)*(log10(x(i,2))-z2A);
    Sz2z2=Sz2z2+(log10(x(i,2))-z2A)*(log10(x(i,2))-z2A);
end;
Szz(1,1)=Sz1z1; Szz(1,2)=Sz1z2;
Szz(2,1)=Sz1z2; Szz(2,2)=Sz2z2;

Szz_1=1/Szz;

zi(1)=log10(x1i)-z1A; zi(2)=log10(x2i)-z2A;
ziT=zi';
ziSzz_1=ziT*Szz_1;
result=ziSzz_1*zi;

Df=n-3;
alpha=1-Pconf;
P=1-alpha/2;
//P=0.975;
Q=1-P;

[T]=cdfst("T",Df,P,Q);

b0=-0.05477;
b1=0.67226;
b2=0.44154;

```

```

Szyzyn4=0; Syy=0;
for i=1:n
    Szyzyn4=Szyzyn4+...
        (b0+b1*log10(x(i,1))+b2*log10(x(i,2))-log10(yx(i,1)))^2;
    Syy=Syy+(yx(i,1)-10^b0*x(i,1)^b1*x(i,2)^b2)^2;
end;
Szyzyn4=Szyzyn4/(n-3)
Delta_z=T*(Szyzyn4*(1/n+result))^0.5;
Delta_z_pr=T*(Szyzyn4*(1+1/n+result))^0.5;

r_2=1-Syy/SyA;

zy_result=b0+b1*log10(x1i)+b2*log10(x2i);

zyLBC=zy_result-Delta_z; zyUBC=zy_result+Delta_z;

zyLBPr=zy_result-Delta_z_pr; zyUBPr=zy_result+Delta_z_pr;

y_result=10^zy_result;
yLBC=10^zyLBC; yUBC=10^zyUBC;
yLBPr=10^zyLBPr; yUBPr=10^zyUBPr;

file_PC5=fopen('D:\Data_F\Web_Django_30_data5.txt','w+');
    fprintf(file_PC5, "%5.6f ", yUBC);
fclose(file_PC5);

file_PC4=fopen('D:\Data_F\Web_Django_30_data4.txt','w+');
for i = 1:2
    for j = 1:2 fprintf(file_PC4, "%5.6f ", Szz_1(i,j)); end;
    fprintf(file_PC4, "\n");
end;
fclose(file_PC4);

set(textresult,'string', sprintf("=%1.4f",y_result));
set(textresult2,'string', sprintf("=%1.4f",yLBC));
set(textresult3,'string', sprintf("=%1.4f",yUBC));
set(textresult4,'string',...
    sprintf("Коефіцієнт детермінації R^2=%1.4f",r_2));
set(textresult5,'string', sprintf("=%1.4f",yLBPr));
set(textresult6,'string', sprintf("=%1.4f",yUBPr));

if (get radio_bikv_x,'value')==0)

else

```

```

yfact=eval(get(edit_text12,'string'));
MRE=abs((yfact-y_result)/yfact);

textresult5=uicontrol(f,'style','text','string','', 'position',...
[500,120,160,20]);
set(textresult5,'string',...
sprintf("Відносна похибка MRE=%1.4f",MRE));
end;

endfunction

```

Далі в середовищі Scilab 6.0.0 було створено sce-файл розміром 14 КБ. Зазначимо, що в програмі використовувався динамічний спосіб створення інтерфейсних компонентів. Він полягає у тому, що при виконанні програми можуть створюватися і знищуватися ті чи інші елементи управління (командні кнопки, радіокнопки, мітки та таке інше), а їх властивостям присвоюються відповідні значення. В програмі були використані такі інтерфейсні компоненти: вікно редагування, командні кнопки та радіокнопки. Графічне вікно програми із зазначеними інтерфейсними компонентами наведено на рис.3.4.

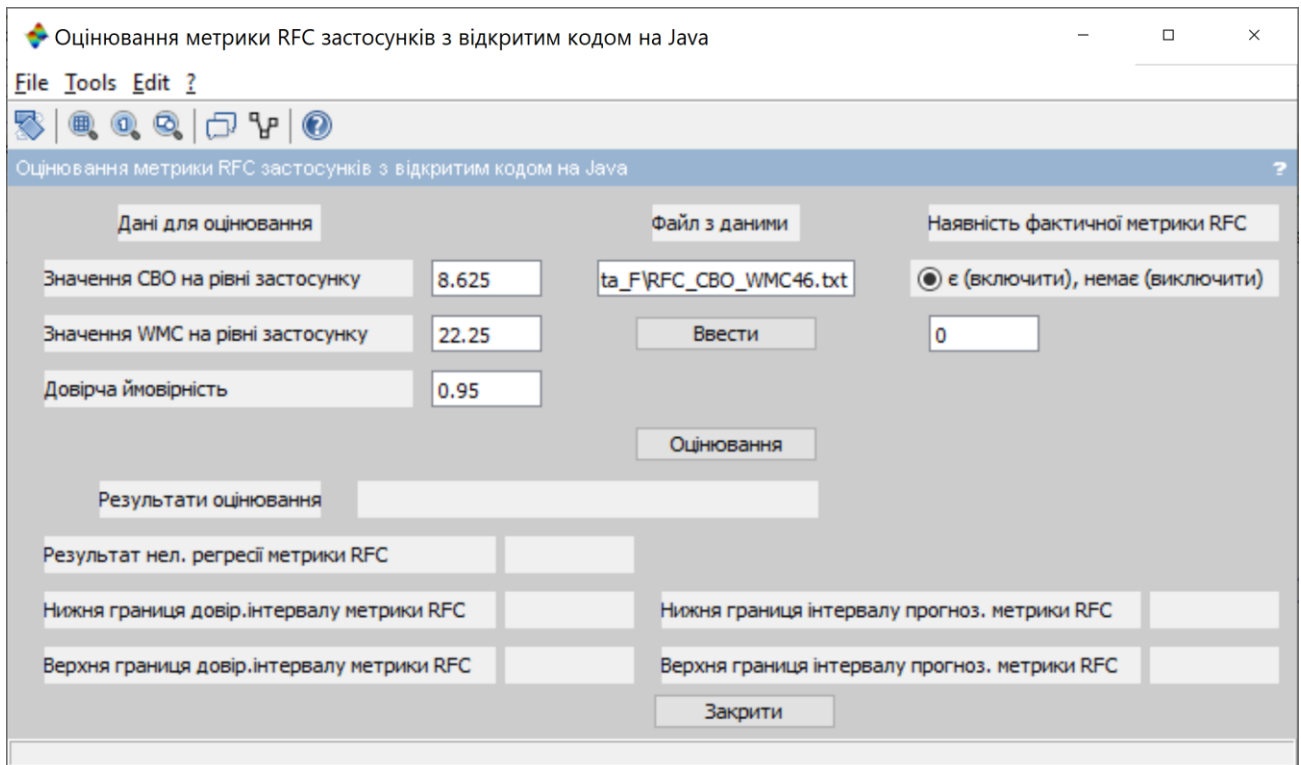


Рисунок 3.4. – Графічне вікно програми із інтерфейсними компонентами

Було проведено тестування окремих модулів. У якості прикладу наведемо результати тестування для модуля оцінювання метрики RFC застосунків з відкритим кодом на Java. Було побудовано нелінійну регресійну модель на основі нормалізуючого перетворення у формі десяткового логарифму. Результати візуалізації емпіричних даних з метрики RFC 46 застосунків з відкритим кодом на Java в залежності від метрики CBO наведені на рис.3.5.

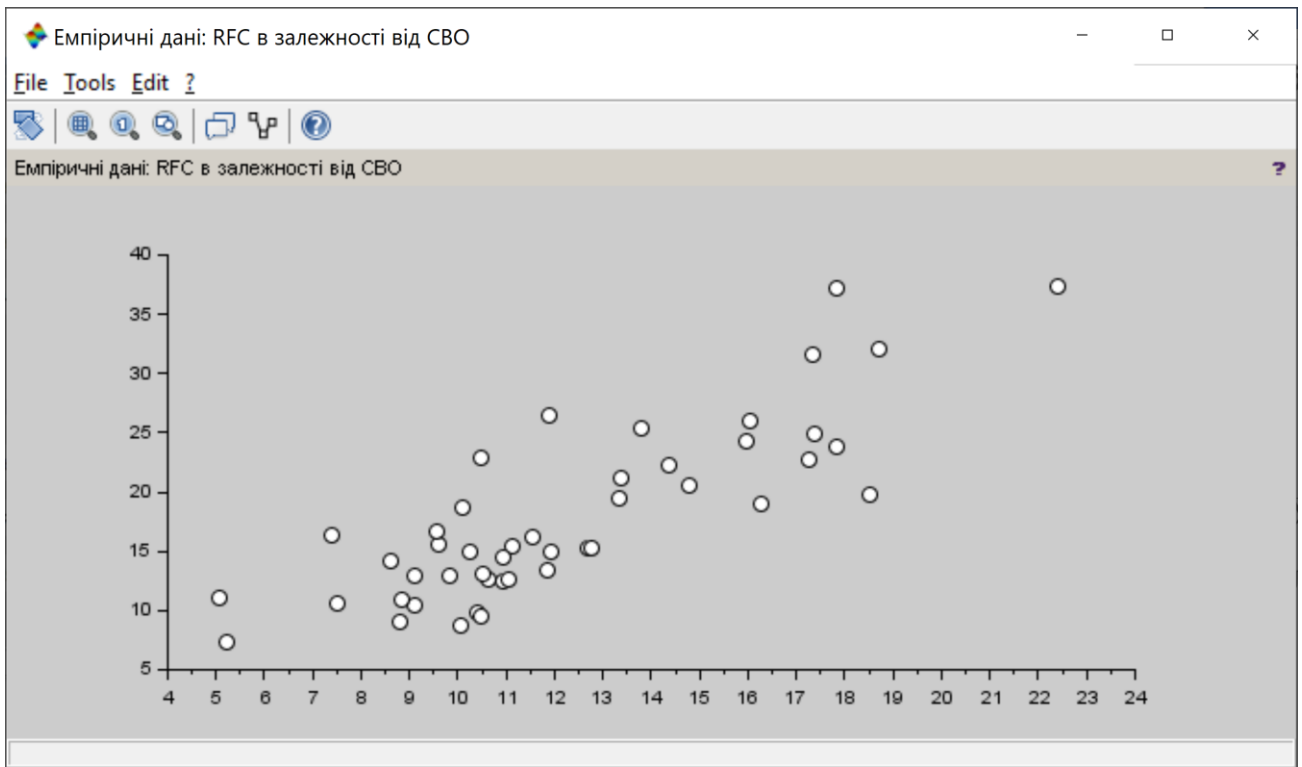


Рисунок 3.5. – Емпіричні дані з метрики RFC 46 застосунків з відкритим кодом на Java в залежності від метрики CBO

Результати візуалізації емпіричних даних з метрики RFC 46 застосунків з відкритим кодом на Java в залежності від метрики WMC наведені на рис.3.6.

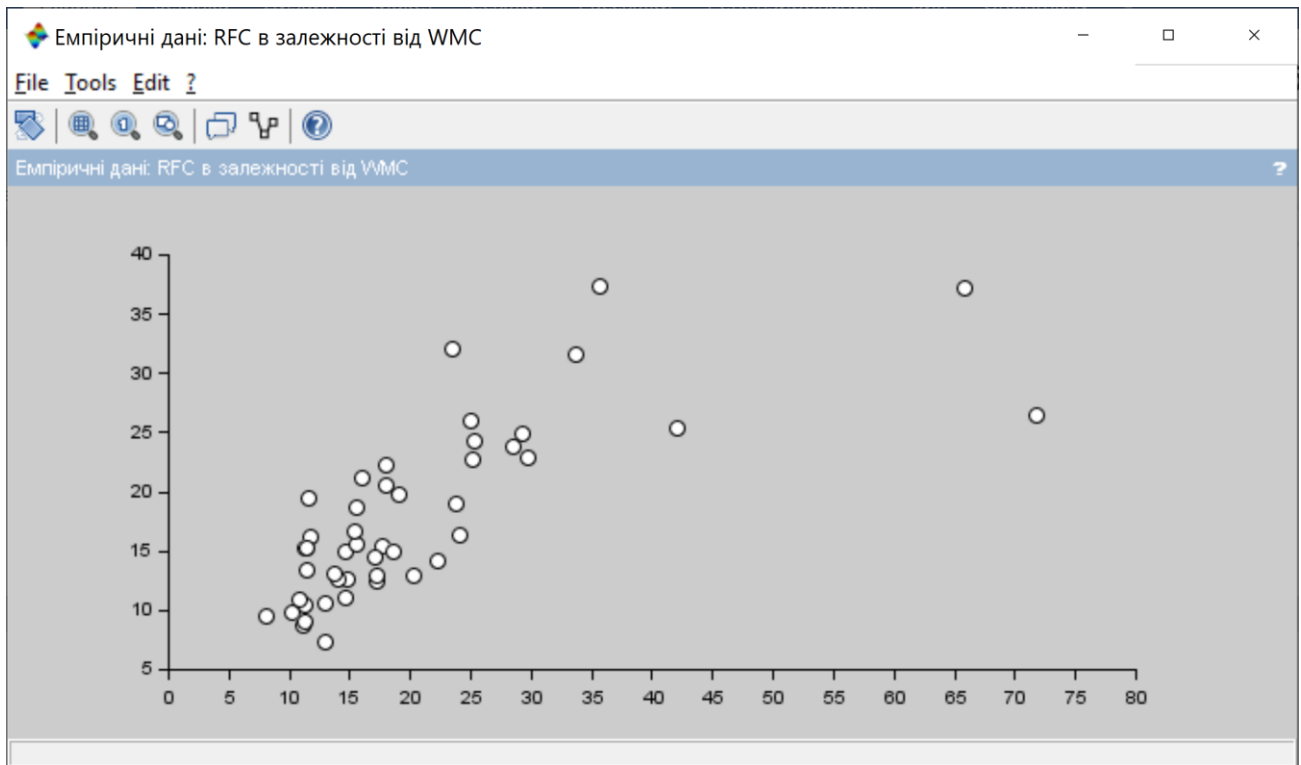


Рисунок 3.6. – Емпіричні дані з метрики RFC 46 застосунків з відкритим кодом на Java в залежності від метрики WMC

Результати оцінювання значень нелінійної регресії, нижньої і верхньої границь довірчого інтервалу нелінійної регресії метрики RFC застосунків з відкритим кодом на Java, та результати обчислення коефіцієнту детермінації R^2 наведені на рис.3.7. Також у разі відомого фактичного значення метрики RFC застосунку може бути обчислено значення відносної похибки (MRE) результату оцінювання значень нелінійної регресії в порівнянні з фактичним значення метрики RFC для зазначених метрик CBO та WMC.

Оцінювання метрики RFC застосунків з відкритим кодом на Java

File Tools Edit ?

Оцінювання метрики RFC застосунків з відкритим кодом на Java

Дані для оцінювання		Файл з даними	Наявність фактичної метрики RFC	
Значення CBO на рівні застосунку	8.625	ta_F\RFC_CBO_WMC46.txt	☐ є (включити), немає (виключити)	
Значення WMC на рівні застосунку	22.25	Ввести	0	
Довірча ймовірність	0.95			
Оцінювання				
Результати оцінювання		Коефіцієнт детермінації $R^2=0.8718$		
Результат нел. регресії метрики RFC	=14.7640			
Нижня границя довір.інтервалу метрики RFC	=13.6332	Нижня границя інтервалу прогноз. метрики RFC	=10.7140	
Верхня границя довір.інтервалу метрики RFC	=15.9886	Верхня границя інтервалу прогноз. метрики RFC	=20.3450	
Закрити				

Рисунок 3.7. – Результати оцінювання значень нелінійної регресії, нижньої і верхньої границь довірчого інтервалу нелінійної регресії метрики RFC застосунку з відкритим кодом на Java

На рис.3.8 наведено значення відносної похибки (MRE) результату оцінювання значення нелінійної регресії в порівнянні з фактичним значенням метрики RFC застосунку для зазначених метрик CBO та WMC.

Оцінювання метрики RFC застосунків з відкритим кодом на Java

File Tools Edit ?

Оцінювання метрики RFC застосунків з відкритим кодом на Java

Дані для оцінювання		Файл з даними	Наявність фактичної метрики RFC	
Значення CBO на рівні застосунку	8.625	ta_F\RFC_CBO_WMC46.txt	☒ є (включити), немає (виключити)	
Значення WMC на рівні застосунку	22.25	Ввести	14.125	
Довірча ймовірність	0.95			
Оцінювання				
Результати оцінювання		Коефіцієнт детермінації $R^2=0.8718$		Відносна похибка MRE=0.0452
Результат нел. регресії метрики RFC	=14.7640			
Нижня границя довір.інтервалу метрики RFC	=13.6332	Нижня границя інтервалу прогноз. метрики RFC	=10.7140	
Верхня границя довір.інтервалу метрики RFC	=15.9886	Верхня границя інтервалу прогноз. метрики RFC	=20.3450	
Закрити				

Рисунок 3.8. – Результати визначення відносної похибки оцінювання значення нелінійної регресії в порівнянні з фактичним значенням метрики RFC застосунку для зазначених метрик CBO та WMC

Отримані результати практично співпадають з результатами, що наведені у табл. 2.1 та оцінками інших авторів. Це свідчить про працездатність розробленої програми.

3.5 Висновки за розділом 3

У даному розділі виконано постановку задачі на розробку проєкту програми для оцінювання метрики RFC застосунків з відкритим кодом на Java, розроблено ескізний, технічний та робочий проєкти програми для оцінювання метрики RFC застосунків з відкритим кодом на Java.

1) В ескізному проєкті було виявлено варіанти використання та побудовано їх сценарії, розроблено зовнішню специфікацію програми для оцінювання метрики RFC застосунків з відкритим кодом на Java та стани цієї програми при виконанні.

2) В технічному проєкті було спроектовано програму для оцінювання метрики RFC застосунків з відкритим кодом на Java на основі покрокової деталізації зверху вниз, розроблено специфікації модулів програми, побудовано алгоритми модулів програми.

3) В робочому проєкті було розроблено програму для оцінювання метрики RFC застосунків з відкритим кодом на Java здійснено тестування програми, розроблено програмну документацію, здійснено випробування програми. Результати випробування програми практично співпадають з результатами, що отримані іншими авторами. Це свідчить про працездатність розробленої програми.

4 РОЗРАХУНОК ЕКОНОМІЧНОЇ ЕФЕКТИВНОСТІ ВПРОВАДЖЕННЯ ПРОГРАМИ

Найбільш важливим моментом при проектуванні комп'ютерної програми для розробника, з економічної точки зору, є процес формування її вартості. Вочевидь, що комп'ютерна програма являє собою дуже специфічний товар з безліччю особливостей.

Створення програми вимагає одноразових витрат на її розробку, придбання необхідних технічних засобів і поточних витрат на функціонування продукту. Економія від функціонування програми визначається з урахуванням витрат на її експлуатацію. Відношення цієї економії до витрат на створення програми характеризує економічну ефективність капітальних вкладень. Економічні показники визначаються по діючим на момент розрахунку оптовим цінам, тарифам і ставкам заробітної плати.

4.1 Розрахунок витрат на створення й експлуатацію програми

Витрати на розробку програми складаються з витрат на зарплату розробника, на амортизацію комп'ютера, на якому виконується розробка, на експлуатацію цього комп'ютера, на засоби розробки та витрат на матеріали і комплектуючі.

Розробка програми виконується програмістом, місячний оклад якого складає 30000 грн. Додаткова заробітна плата складає 20% від основної. Виходячи з цього, основна і додаткова заробітна плата програміста

36000 грн/мес, а вартість сучасного ноутбука складає 26076 грн. (приведена вартість ноутбука Lenovo IdeaPad 3 15ALC6 (82KU00NERA) Arctic Grey: Екран 15.6" TN (1920x1080) Full HD, матовий / AMD Ryzen 3 5300U (2.6 - 3.8 ГГц) / RAM 8 ГБ / SSD 128 ГБ / AMD Radeon Graphics / без ОД / Wi-Fi / Bluetooth / веб-камера / Windows 11 Професійна на 1ПК (ESD - електронна ліцензія в конверті, всі мови) (FQC-10572)). При вартості кіловат-години електроенергії у 1,7166 грн. (II клас напруги до 27,5 кВт, разом з ПДВ), розраховується вартість розробки ПЗ. Витрати на допоміжні матеріали приведені в табл. 4.1

Таблиця 4.1 – Витрати на допоміжні матеріали

Пункти витрат	Сума
Папір	180,00
Заправлення картриджа до принтера	180,00
CD-диск	12,00
Непередбачені витрати	300,00
Разом матеріали і комплектуючі	672,00

Вартість програми розраховуємо за формулою:

$$\text{Спр} = (\text{Ззп} + \text{Зсз} + \text{Ззг} + \text{Зе}) * \text{T} + \text{Зм}, \quad (4.1)$$

де Т – тривалість розробки, міс.; Ззп – основна і додаткова заробітна плата обслуговуючого персоналу, грн.; Зсз – відрахування на соціальні заходи або єдиний соціальний внесок (36,76% від основної і додаткової заробітної плати), грн.; Ззг – загальногосподарські витрати (10% від основної заробітної плати), грн.; Зе – витрати на електроенергію; Зм – витрати на основні і допоміжні матеріали. Зе при споживанні потужності 0,5 кВт, тривалості роботи за місяць, рівної $22 * 8 = 176$ годин, вартості кіловат-години електроенергії 1,7166 грн. складає

$$З_е = 176 * 1,7166 * 0,5 = 151,06 \text{ грн.}$$

Таблиця 4.2 – Витрати на розробку ПЗ

Найменування витрат	Одиниця	Кількість
Тривалість розробки	Мес	1,5
Основна і додаткова ЗП	Грн.	36000
Відрахування на соціальні заходи	Грн.	13233,6
Загальногосподарські витрати	Грн.	500
Витрати на матеріали	Грн.	672,00
Витрати на електроенергію	Грн.	151,06

Відповідно до формули (4.1) вартість програми:

$$\text{Спр} = (36000 + 13233,6 + 500 + 151,06) * 1,5 + 672,00 = 75498,99 \text{ грн.}$$

Амортизаційні відрахування на устаткування складають 60% балансової вартості в рік:

$$\text{Аоб} = 26076 * 0,6 = 15645,6 \text{ грн.}$$

У масштабах підприємства річні витрати на основні і допоміжні матеріали (гнучкі диски, папір) визначаються в розмірі 5% вартості основного устаткування:

$$\text{Вм} = 16800 * 0,05 = 1303,8 \text{ грн.}$$

Річний обсяг робіт ПК у годинах визначається в такий спосіб:

$$\text{Фм} = 264,5 * \text{Тз}, \quad (4.2)$$

де Тз – це середнє місячне завантаження устаткування (близько 4 годин); 264,5 – середня кількість робочих днів у році.

Отже, річний обсяг роботи ПК складе:

$$\Phi_m = 264,5 \cdot 4 = 1058 \text{ годин.}$$

Витрати на електроенергію Z_e при 1013 годинах роботи устаткування в рік складуть

$$Z_e = 1058 \cdot 0,5 \cdot 1,7166 = 908,08 \text{ грн.}$$

Експлуатаційні витрати для ПК за рік складуть:

$$Z_{zp} = 15645,6 + 1303,8 + 908,08 = 17857,48 \text{ грн.}$$

Отже, у перший рік витрати на створення й експлуатацію програми складуть:

$$Z_{se} = 75498,99 + 17857,48 = 93356,47 \text{ грн.}$$

4.2 Економічна ефективність розробки і впровадження програми

Основним показником економічної ефективності функціонування програми є підвищення ефективності керування інформацією у вигляді зниження витрат на керування при одночасному збільшенні швидкості і якості одержання потрібного результату.

Крім багатьох інших негативних ефектів, ручна обробка інформації спричиняє наступні негативні економічні ефекти:

- високі витрати на складування паперових документів (сейфи, шафи, папки й ін.);
- підвищені витрати на канцтовари;

– витрати, пов'язані з роботою виявлення раніше допущених помилок (людський фактор).

До числа основних факторів, що визначають приріст прибутку в зв'язку з упровадженням програми, відносяться:

- підвищення продуктивності праці;
- вивільнення робочого часу.

Крім того, не піддається прямій грошовій оцінці підвищення оперативності керування, якість одержуваних результатів, поліпшення організації праці і т.д.

Обов'язковою умовою визначення економічної ефективності програми є порівнянність усіх показників у часі, за цінами й іншими нормами, використовуваними для визначення показників, за змістом і колом елементів витрат.

Визначимо пряму економічну ефективність, ґрунтуючись на тому, що впровадження програмного продукту вивільняє 0,6 працівника (за експертною оцінкою фахівців підприємства).

Зарплата 0,6 працівника в рік складає:

$$(30000 \cdot 12) \cdot 0,6 = 216000 \text{ грн.}$$

Річний економічний ефект розраховується по формулі:

$$E_p = \Delta C_n - E_n \cdot k,$$

де ΔC_n – вивільнені кошти після впровадження програми (216000 грн.) мінус експлуатаційні витрати (17857,48 грн.); E_n – коефіцієнт ефективності (дорівнює коефіцієнту амортизації – 0,6); k – одноразові витрати на впровадження програми (75498,99 грн.).

Річний економічний ефект буде

$$E_p = 198142,52 - 45299,39 = 152843,13 \text{ грн.}$$

Строк окупності програми розраховується по формулі:

$$T = \frac{k}{\Delta C_n} = \frac{75498,99}{198142,52} = 0,38 \text{ (року).}$$

Отже строк окупності програми складає приблизно 4,6 місяця.

5 ОХОРОНА ПРАЦІ

5.1 Аналіз існуючої проблеми охорони праці

Охорона праці – це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних та лікувально-профілактичних заходів та засобів, спрямованих на збереження здоров'я і працездатності людей в процесі праці (ст. 1 Закону «Про охорону праці» [18]). Як правовий інститут охорона праці – це чималий комплекс правових норм. Можна визначити охорону праці також як систему забезпечення безпеки, збереження здоров'я і працездатності людини в процесі праці, засновану на сукупності законодавчих актів і відповідних їм соціально-економічних, технічних, гігієнічних і організаційних заходів.

Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці.

Власник або уповноважений ним орган повинен впроваджувати сучасні засоби техніки безпеки, які запобігають виробничому травматизму, і забезпечувати санітарно-гігієнічні умови, що запобігають виникненню професійних захворювань працівників.

На власника або уповноважений ним орган покладається систематичне проведення інструктажу (навчання) працівників з питань охорони праці, протипожежної охорони.

Охорона праці – один з центральних інститутів трудового права. Він має виключно практичне значення. Недодержання вимог охорони праці створює небезпеку для здоров'я і життя працівників. У свою чергу і ті, кого законодавець називає власником або уповноваженим ним органом, несуть сувору, у тому числі й кримінальну відповідальність за порушення правил охорони праці.

5.2 Аналіз шкідливих та небезпечних факторів в офісі фірми з персональними комп'ютерами

Для аналізу шкідливих та небезпечних факторів необхідно розглянути характеристики умов праці робітників, а також можливі фактори, що можуть негативно вплинути на їхнє здоров'я.

Під умовами праці розуміють сукупність факторів та обставин трудового процесу і виробничого середовища, в якому здійснюється діяльність людини, що впливають на здоров'я та працездатність.

Під факторами трудового процесу розуміють основні його характеристики: важкість праці та напруженість праці. Важкість праці – це характеристика трудового процесу, яка відображає переважне навантаження на опорно-руховий апарат і функціональні системи організму (серцево-судинну, дихальну та інші, що забезпечують його діяльність). Напруженість праці – це характеристика трудового процесу, яка відображає навантаження переважно на центральну нервову систему, органи чуття, емоційну сферу працівника.

Під виробничим середовищем розуміють сукупність фізичних, хімічних, біологічних, психофізіологічних факторів на виробництві, що діють на людину. Усі ці фактори класифікують як небезпечні та шкідливі.

Небезпечні виробничі фактори – це ті, вплив яких на працівника призводить до травм, раптового погіршення здоров'я чи до смерті.

Шкідливі виробничі фактори – це ті, вплив яких на працівника може призвести до захворювання та зниження працездатності.

До фізичних небезпечних і шкідливих факторів належать:

- рухомі машини і механізми, рухомі частини виробничого обладнання, вироби, що пересуваються (матеріали, заготовки);
- підвищена запиленість і загазованість повітря робочої зони;
- підвищена чи знижена температура поверхонь обладнання, матеріалів, повітря робочої зони;
- підвищені рівні шуму, вібрації, ультразвуку, інфразвукових коливань;
- підвищений чи знижений барометричний тиск і його різкі зміни;
- підвищена чи знижена вологість, рухомість, іонізація повітря;
- підвищений рівень іонізуючих випромінювань, напруги в електромережі, статичної електрики, електромагнітних випромінювань, напруженості електричного і магнітного полів;
- відсутність чи брак природного світла, знижена контрастність, пряма і відбита блискотливість, підвищена пульсація світлового потоку;
- підвищені рівні ультрафіолетової та інфрачервоної радіації;
- гострі краї, шершавість на поверхні заготовок, інструментів і обладнання;
- розташування робочого місця на значній висоті відносно землі (підлоги).

До хімічних небезпечних і шкідливих виробничих факторів належать речовини, які за характером впливу на організм людини поділяються на токсичні, подразнюючі, сенсабілізуючі, канцерогенні і мутагенні. Ці хімічні речовини впливають на репродуктивну функцію людини. За шляхами проникнення в організм людини вони поділяються на ті, що проникають через органи дихання, шлунково-кишковий тракт, шкіряний покрив і слизові оболонки.

До біологічних небезпечних і шкідливих виробничих факторів належать патогенні мікроорганізми (бактерії, віруси, рикетсії, спірохети, грибки) та продукти їх життєдіяльності, а також рослини та живі істоти.

До психофізіологічних небезпечних і шкідливих виробничих факторів належать фізичні (статичні і динамічні) та нервово-психічні перевантаження (розумове перенапруження, перенапруження аналізаторів, монотонність праці, емоційні перевантаження).

До небезпечних факторів стосується і гігієнічні умови праці робочих місць, що негативно впливають на здоров'я людей, які там працюють.

Гігієнічне оцінювання умов і характеру праці на робочих місцях виконується на основі гігієнічної класифікації праці за показниками шкідливості та небезпечності факторів виробничого середовища, важкості та напруженості трудового процесу. Гігієнічна класифікація базується на принципі диференціації умов праці залежно від фактично визначених рівнів указаних факторів у порівнянні з санітарними нормами, правилами, гігієнічними нормами, а також з урахуванням можливого шкідливого впливу їх на стан здоров'я працівників.

Умови праці та принципи гігієнічної класифікації поділяються на чотири класи.

Оптимальні умови праці (1-й клас) – умови, за яких зберігається не лише здоров'я працівників, а й створюються передумови для підтримання високого рівня працездатності. Оптимальні гігієнічні нормативи виробничих факторів встановлені для мікроклімату і факторів трудового процесу. Для інших факторів за оптимальні умовно приймаються такі умови праці, за яких несприятливі фактори виробничого середовища не перевищують рівнів, прийнятих за безпечні для населення.

Допустимі умови праці (2-й клас) – характеризуються такими рівнями шкідливих виробничих факторів виробничого середовища і трудового процесу, які не перевищують встановлених гігієнічних нормативів, а можливі зміни функціонального стану організму відновлюються за час регламентованого відпочинку чи до початку наступної зміни та не чинять несприятливого впливу на стан здоров'я працівників та їх потомство в найближчому і віддаленому періодах.

Шкідливі умови праці (3-й клас) – характеризуються такими рівнями шкідливих виробничих факторів, які перевищують гігієнічні нормативи і здатні несприятливо впливати на організм працівника та на його нащадків. Шкідливі умови праці за ступенем перевищення гігієнічних нормативів і вираженості можливих змін в організмі працівників поділяються на чотири ступені.

Перший ступінь – умови праці характеризуються таким рівнем шкідливих факторів виробничого середовища і трудового процесу, які, як правило, зумовлюють функціональні зміни, що виходять за межі фізіологічних коливань (останні відновлюються при тривалішій, ніж початок наступної зміни, перерві контакту зі шкідливими факторами) та збільшують ризик погіршення здоров'я.

Другий ступінь – умови праці характеризуються такими рівнями шкідливих факторів виробничого середовища і трудового процесу, які здатні

спричинювати стійкі функціональні порушення, призводять у більшості випадків до зростання виробничо-зумовленої захворюваності, появи окремих ознак або легких форм професійної патології (як правило, без втрати професійної працездатності), що виникають після тривалої експозиції (10 років і більше).

Третій ступінь – умови праці характеризуються такими рівнями шкідливих факторів виробничого середовища і трудового процесу, які призводять, окрім зростання виробничо-зумовленої захворюваності, до розвитку професійних захворювань, як правило, легкого і середнього ступенів важкості (з утратою професійної працездатності в період трудової діяльності).

Четвертий ступінь – умови праці характеризуються такими рівнями шкідливих факторів виробничого середовища і трудового процесу, які здатні призводити до значного зростання хронічної патології та рівнів захворюваності з тимчасовою втратою працездатності, а також важких форм професійних захворювань (з утратою загальної працездатності).

Небезпечні (екстремальні) умови праці (4-й клас) – умови праці характеризуються такими рівнями шкідливих факторів виробничого середовища і трудового процесу, вплив яких протягом робочої зміни (чи її частини) створює загрозу для життя, високий ризик виникнення важких форм професійних уражень.

Поширена думка, що офісні робітники працюють в приємній, безпечній обстановці. Хоча працювати в офісі не так небезпечно, як у багатьох інших місцях, тут також існує ряд джерел ризику для життя і здоров'я. Деякі з них представляють для людей, які працюють в офісі серйозну небезпеку.

Для працівника несприятливими подіями внаслідок впливу умов праці є втома, стрес, захворювання (хвороба) або травма.

Втома – це фізіологічний стан організму, що виникає внаслідок надмірно інтенсивного або тривалої діяльності, що виявляється тимчасовим зниженням функціональних можливостей людського організму. Розрізняють фізичну, розумову і емоційну втому.

Фізична втома проявляється порушенням функції м'язів: зниженням сили, точності, узгодженості і ритмічності рухів; виникає при інтенсивній або тривалій фізичній діяльності.

Розумова втома проявляється зниженням продуктивності інтелектуальної праці, ослабленням уваги (труднощі зосередження), уповільненням мислення, зниженням показників розумової активності, інтересу до роботи; виникає при інтенсивній інтелектуальній діяльності.

Емоційна втома проявляється помітним зниженням емоційних реакцій під впливом надсильних або монотонних подразників (стресів).

Надмірне робоче навантаження протягом тривалого часу або недостатній час відпочинку може привести до хронічної втоми або перевтоми. Розрізняють розумовий і психічний (душевний) перевтома. В умовах сучасного ритму праці і життя все частіше у працівників з'являється синдром хронічної втоми.

Стрес є серйозною проблемою соціально-психологічного характеру для багатьох організацій. Він може бути викликаний багатьма факторами, включаючи шум від переповнення приміщення людьми або працюючого обладнання, поганих відносин з начальством та / або колегами, збільшення робочого навантаження і недостатньою свободою дій при виконанні роботи.

Іншим масово поширеним несприятливим наслідком праці є те або інше захворювання: нездужання, погане самопочуття, симптоми, які можуть проходити бурхливо або відносно швидко минути («гострі» -з медичної термінології) або тривати (роками) або періодично загострюватися (хронічні).

Захворювання, спровоковані умовами праці, часто називають виробничозумовленими захворюваннями.

Специфічний вплив факторів, пов'язаний з конкретними виробничими чинниками, призводить до розвитку певних, викликаних цими факторами, захворювань. Оскільки вони викликані несприятливими умовами праці конкретних робочих місць конкретних професій, то їх називають професійними захворюваннями.

Захворювання кістково-м'язової системи та пошкодження м'яких тканин типу тендиніту виникають в результаті використання офісних меблів та обладнання, які не відповідають індивідуальним анатомічним особливостям. Тендинит може розвинутися в результаті повторюваних рухів окремих частин тіла. Так, наприклад, у службовців починають боліти пальці від тривалого писання або при роботі з дуже товстими папками, які доводиться постійно діставати із шаф і повертати на місце. Багато офісних робітників страждають від різноманітних RSI, типу захворювань зап'ястя і грудної клітини, через погану підгонку меблів і устаткування і відсутності перерв у друкуванні (на клавіатурі комп'ютера) або інших періодично повторюваних діях. Недосконала конструкція меблів і устаткування призводять також до порушення постави і здавлення нервів нижніх кінцівок, тому що багато офісних робітників велику частину часу 87 проводять сидячи. Всі ці фактори сприяють виникненню захворювань попереку і нижніх кінцівок в такій же мірі, як і тривале стояння на ногах.

Безперервне використання комп'ютерів і погане загальне освітлення викликає напругу зору, в результаті чого воно поступово погіршується, з'являються головні болі, печіння і відчуття втоми в очах. Регулювання рівня освітленості приміщення і контрастності зображення на екрані комп'ютера, а

також часта перефокусування очей необхідні для попередження виникнення проблем із зором. Освітлення має відповідати характеру виконуваної роботи.

5.3 Розрахунок рівня освітлення в офісі фірми

Рівень освітленості впливає на працездатність і самопочуття людини. Державні норми це враховують і визначають, яким має бути освітлення приміщень, залежно від їх призначення. Відштовхуючись від цих норм, планують освітлення об'єктів, а отже, від них безпосередньо залежить і те, які світильники і скільки лампочок необхідно купити.

Стосовно ДБН (Державно будівельні норми) є конкретні показники якості освітлення приміщень різних призначень: житлових, офісних, складських, виробничих. Спираючись на ці дані, розраховують рівень освітленості, враховуючи площу приміщення, висоту стелі, відбиваючу здатність поверхонь, тип світильника та інші параметри. Стосовно необхідного нам показника в офісних приміщеннях, нормою освітлення буде 300 Лк.

Норми виникли після того як були враховані всі найбільш важливі особливості фізіологічних зорових процесів людини і фактори, які впливають на концентрацію, зорову напругу під час роботи, відпочинку (наприклад, контрастність, колір робочої поверхні).

Для розрахунку рівня освітленості приміщення зазвичай використовується спосіб, який передбачає використання даних про величину світлового потоку, який вимірюється люменами (Лм). Світловий потік, що випромінює лампа, який вказується на упаковці придбаної лампи.

Для визначення величини світлового потоку використовується формула:

Норма освітлення * Площа * Коефіцієнт висоти стелі.

При висоті стелі від 2,5 м до 2,7 м коефіцієнт дорівнює 1, при висоті 2,7-3 метрів - 1,2, якщо висота становить від 3 до 3,5 метрів - 1,5, від 3,5 м до 4,5 м - 2. Порахуємо, яка величина світлового потоку необхідна вітальні площею 30 квадратних метрів з висотою стель 2,5 м:

$$150 * 30 * 1 = 4500 \text{ (Лм)}.$$

Оскільки різні лампочки випромінюють світло різної сабо, потрібна їх кількість залежить від того, що ми виберемо – світлодіодну, енергозберігаючу, галогенну або лампу розжарювання, а також від того, якої потужності вона буде.

Таблиця 5.1 – Приклад різних видів ламп і їх потужності

Лампочка	Потужність, Вт	Світловий потік, Лм	Потрібно лампочок, шт
Розжарювання	40	470	$4500/470=9,5$
Галогенна	32	470	$4500/470=9,5$
Енергозберігаюча	15	700	$4500/700=6,4$
Світлодіодна	10	750	$4500/750=6$

Отже, для освітлення вітальні площею 30 квадратних метрів необхідно 10 галогенних ламп потужністю 30 Вт, стільки ж більш потужних лампочок розжарювання, 7 енергозберігаючих по 15 Вт кожна або 6 світлодіодних 10- 89 ватних лампочок. Після підрахунків підсумок з десятковими знаками завжди округляємо в більшу сторону.

Більш складний спосіб розрахунку включає більше змінних:

$N = (E * S) / (U * p * Fi * Kз)$, де N – кількість світильників, E – норма освітленості, S – площа об'єкта, $Kз$ – коефіцієнт запасу (залежить від виду лампочки, ступеня забруднення приміщення), U – коефіцієнт використання світлового потоку (залежить від висоти стін, площі кімнати, типу світильника, кольору покриття стелі, стін, підлоги), p – кількість патронів світильника, Fi – значення світлового потоку однієї лампи.

Наприклад, для вітальні з першого прикладу планується встановити точкові світильники. Колір покриття стелі - білий, стіни світлі, на підлозі - темно-коричневий ламінат. Для об'єкта показник E становить 150 (дивимося в таблиці), S – 30, $Kз$ – 1 (припускаємо, що купують світлодіодні лампи потужністю 10 Вт), U – 0,46, p – 1, Fi – 750.

Підставляємо дані:

$$N = (150 * 30) / (0,46 * 1 * 750 * 1) = 4500 / 345 = 13.$$

Виходячи зі розрахованої формули було знайдено оптимальне значення кількості світильників, тобто слід розмістити 13 світильників.

5.4 Розробка заходів щодо зменшення впливу шкідливих та небезпечних факторів

З метою запобігання або зменшення впливу шкідливих і небезпечних виробничих чинників під час підготовки робочих місць, або робочої зони, необхідно слідувати загальним вимогам, що встановлені законодавством України про працю.

Відповідно до законів України «Про охорону праці» умови праці на робочому місці, безпека технологічних процесів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам законодавства.

Більшість нормативів щодо умов праці офісних працівників встановлено на рівні державних стандартів. Відповідно до ч. 1 ст. 13 Закону про охорону праці роботодавець зобов'язаний створити на робочому місці в кожному структурному підрозділі умови праці відповідно до нормативно-правових актів.

Як відомо, тривала робота за комп'ютером та з документами при недостатньому рівні освітленості може призвести до значного перенапруження зору, тому вимоги до освітлення є досить важливими. Додатково, окрім вже перелічених документів, вимоги до освітлення встановлено ДБН В.2.5-28-2006 «Природне і штучне освітлення», затвердженими наказом Мінрегіону від 15.05.2006 р. № 168.

Як вже зазначалося, відносно вікон робоче місце необхідно організовувати так, щоб природне світло було з лівого боку (п. 4.3 ДСанПіН 3.3.2.007-98). Робоче місце необхідно розміщувати таким чином, щоб уникнути попадання прямого світла в очі. Для забезпечення захисту і досягнення нормованих рівнів комп'ютерних випромінювань необхідне застосування приєкраних фільтрів, локальних світлофільтрів (засобів індивідуального захисту очей) та інших засобів захисту, що пройшли випробування в акредитованих лабораторіях і мають щорічний гігієнічний сертифікат (п. 4.19 ДСанПіН 3.3.2.007-98).

Штучне освітлення приміщення має здійснюватись системою загального рівномірного освітлення (п. 3.2.2 ДСанПіН 3.3.2.007-98). У приміщеннях при

переважній роботі з документами допускається використання системи комбінованого освітлення, тобто встановлення світильників місцевого освітлення додатково до загального.

Як джерела штучного освітлення необхідно використовувати люмінесцентні лампи. Згідно з п. 3.2.5 ДСанПіН 3.3.2.007-98 система загального освітлення має бути у вигляді суцільних або переривчатих ліній світильників, що розташовані збоку від робочих місць (зазвичай ліворуч) паралельно лінії зору працівників.

Допускається застосування ламп розжарювання у світильниках місцевого освітлення та, у разі влаштування відбитого освітлення у виробничих чи адміністративно-громадських приміщеннях, металогалогенних ламп потужністю 250 Вт.

Коефіцієнт пульсації не повинен перевищувати 5 % (п. 3.2.14 ДСанПіН 3.3.2.007-98). Рівень освітленості на робочому столі в зоні розташування документів має бути в межах 300 – 500 лк. Світильники місцевого освітлення слід встановлювати таким чином, щоб не створювати відблисків на поверхні екрана, а освітленість екрана має не перевищувати 300 лк.

Для забезпечення нормованих значень освітленості у приміщеннях відповідно до п. 3.2.15 ДСанПіН 3.3.2.007-98 необхідно мити вікна і світильники не рідше 2 разів на рік, а також своєчасно замінювати лампи, що перегоріли.

6 ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА

6.1 Аналіз існуючої проблеми необхідності охорони навколишнього середовища

Охорона навколишнього природного середовища, раціональне використання природних ресурсів, забезпечення екологічної безпеки життєдіяльності людини – невід'ємна умова сталого економічного та соціального розвитку України.

З цією метою Україна здійснює на своїй території екологічну політику, спрямовану на збереження безпечного для існування живої і неживої природи навколишнього середовища, захисту життя і здоров'я населення від негативного впливу, зумовленого забрудненням навколишнього природного середовища, досягнення гармонійної взаємодії суспільства і природи, охорону, раціональне використання і відтворення природних ресурсів.

Відносини у галузі охорони навколишнього природного середовища в Україні регулюються Законом України № 1268-ХІІ від 26.06.91 р. «Про охорону навколишнього природного середовища» із змінами [19], а також розроблюваними відповідно до нього земельним, водним, лісовим законодавством, законодавством про надра, про охорону атмосферного повітря, про охорону і використання рослинного і тваринного світу та іншим спеціальним законодавством.

Цей Закон визначає правові, економічні та соціальні основи організації охорони навколишнього природного середовища в інтересах нинішнього і майбутніх поколінь.

Завданням законодавства про охорону навколишнього природного середовища є регулювання відносин у галузі охорони, використання і відтворення природних ресурсів, забезпечення екологічної безпеки, запобігання і ліквідації негативного впливу господарської та іншої діяльності на навколишнє природне середовище, збереження природних ресурсів, генетичного фонду живої природи, ландшафтів та інших природних комплексів, унікальних територій та природних об'єктів, пов'язаних з історико-культурною спадщиною.

Охорона навколишнього природного середовища є системою державних і суспільних заходів, направлених на збереження, відтворювання і раціональне використання природних ресурсів і поліпшення полягання природного середовища, і є частиною прикладної екології.

В міру прискорення темпів науково-технічного прогресу дія людей на природу стає все більш могутньою. І в даний час воно вже сумирно із дією природних чинників, що приводить до якісної зміни співвідношення сил між суспільством і природою. На сучасному етапі людство поставлено перед чинником виникнення в природі незворотних процесів, нових шляхів переміщення і перетворення енергії і речовини. В природу втручається все більше і більше нових речовин, чужих їй, часом сильно токсичних для організмів. Частина з них не включається в природний круговорот і нагромаджується в біосфері, що приводить до небажаних екологічних наслідків.

Накопичення промислових відходів сприяє підвищенню захворюваності людей і тварин, прискоренню корозії машин і металевого устаткування,

зниженню врожайності сільськогосподарських культур і продуктивності тваринництва, прискореному і нераціональному використуванню природних ресурсів і енергії, погіршенню багатьох властивостей екологічних систем, загибелі деяких унікальних природних територіальних комплексів, зникненню окремих видів тварин і рослин.

Основну загрозу для навколишнього середовища в межах галузі розробки ПЗ становлять комплекси електронно-обчислювальних машин (надалі ЕОМ), а саме електромагнітні випромінювання від них.

Дослідження останніх років показали, що електромагнітні випромінювання вищезгаданих електронних пристроїв містять торсіонову компоненту, котра несе інформацію про процеси, що відбуваються в тому чи іншому електронному пристрої.

Торсіонові поля мають високу проникаючу здатність і їх неможливо заекранувати.

Останніми роками при проектуванні (конструюванні), виготовленні (будівництві) і експлуатації технічних і систем управління в різних сферах діяльності надзвичайно широко застосовуються персональні комп'ютери і всілякі комп'ютерні програми.

Робота з комп'ютерами програмістів, операторів і інших користувачів пов'язана з додатковими шкідливими і небезпечними чинниками, що негативно впливають на організм людини. Наприклад, шкідлива дія на зір надає не при тримання стандартних візуальних ергономічних параметрів екрану, розмір мінімального елементу відображення, мерехтіння зображення, відбивна здатність (відблиски) і ін. Працюючий комп'ютер створює електромагнітне поле, що шкідливо діє на організм людини. Це поле може викликати радіоперешкоди, тобто заважати роботі радіо і телеапаратури, що призводить до

зниження надійності технічної системи або системи управління, до збільшення ризику виникнення аварійної ситуації у виробничому середовищі. Для забезпечення безпеки роботи з комп'ютером розроблені і повинні повсюдно застосовуватися стандарти на монітори, вимоги до приміщень для експлуатації комп'ютерів і до організації і устаткування робочих місць.

6.2 Утилізація відходів комп'ютерної техніки

Використання комп'ютерів вимагає вирішення таких важливих питань, як утилізація відходів (мікросхеми з вмістом кольорових металів, плати, дискети).

При утилізації старих комп'ютерів відбувається їх розділ на сім фракцій: метали, пластмаси, штекери, дроти, батареї, скло. Жодна деталь не йде для повторного використання, оскільки не можна гарантувати їх надійність, але у формі вторинної сировини вони йдуть на виготовлення нових комп'ютерів або інших пристроїв.

Детально розглянемо декілька прикладів переробки відходів обчислювальної техніки.

Гадолінієво-галлеві гранати (ГГГ) використовуються у виробництві компонентів пристроїв, що запам'ятовують. В ході обробки біля 80 % вихідного матеріалу перетворюється на відходи або відбраковується. ГГГ мають високу вартість і їх виділення з відходів представляє інтерес з економічної точки зору.

При здобутті досить чистих продуктів можливі повторне їх використання як вихідний матеріал. При цьому значно підвищується економічність виробництва заготовок з ГГГ. Під терміном відходи маються на

увазі кристалічні залишки (залишки середовища для зростання кристалів, частини кристалів, виробництва, що утворюються на різних стадіях), а також дрібний порошок, що виходить при різанні, шліфовці, поліровці кристалів граната або подібних матеріалів.

Переробка цих відходів протягом ряду останніх років викликає труднощі і не вирішена до цих пір. Всі попередні спроби були безуспішними із-за низької розчинності цих складних оксидів.

Вдосконалений процес, розроблений Е. Гуссетом (патент США 4-198231 від 15 квітня 1980 року, фірма «Свісс Алюмініум Лтд.», Швейцарія), призначений для виділення галію і гадолінія з відходів, що містять обидва ці елементи у вигляді оксидів або з'єднань, що перекладаються в оксиди. Відходи дрібно подрібнюються і потім розчиняються в сильних мінеральних кислотах. Гадоліній осідає з очищених розчинів у вигляді оксалата, галій виділяється в металевому вигляді електролітично. Електролітичне виділення галію може проводитися до виділення гадолінія у вигляді оксалата з розчину.

Розглянемо приклад проведення такого процесу. Відходи піддаються переробці, є залишки завантаження в пристрої для зростання кристалів, розколені частини кристалів зі всіх стадій переробки, дрібнозернисті порошки і пудри після операції різання, шліфовки і поліровки гранатів $GdGaO$. Змельчений порошок після обробки кристалів ГГГ висушується при 1200 С і потім нагрівається при 6000 С протягом декількох годин для розкладання летких забруднень.

Дрібнозернисті відходи в кількості 1000 г розміром менше 40 мкм., що містять 34 % галію і 46 % гадолінія кип'ятяться із зворотним холодильником протягом двох годин в 2100 мл 35 %-ной соляної кислоти. Це відповідає 99 % завантаження. Після кип'ятіння частина відходів, що не розчинилася,

фільтрується і промивається. Після висушування залишок важить 20 г. Залишки такого типу об'єднуються і піддаються повторній обробці кип'яченням. Фільтрат, що включає промивні води, об'ємом 2300 мл, обробляється 4000 г металевого галію при 500 С в течії 45 хвилин. Метал має максимально диспергувати.

В ході цієї операції благородніші елементи, присутні в розчині, виділяються і частково розчиняються в галії до його насичення і далі охолоджуються у вигляді інтерметалевих включень або в елементарній формі. Висадження можна проводити з меншою кількістю галію. Метал може періодично замінюватися на нову порцію до досягнення необхідної міри очищення. В результаті процесу очищення виходить розчин з вмістом галію 140 г/л і гадолінія 190 г/л.

Встановлюється величина $ph=1$ шляхом додавання 900 мл 4%-ного розчину перекису водню для окислення домішок заліза. Осадження гадолінія проводиться при 500 С шляхом додавання 1500 г кристалічної технічної щавлевої кислоти $COH*2HO$; суспензія акуратно перемішується 12 годин для повторного осадження у вигляді оксалата гадолінія.

Оксалат гадолінія $Gd(CO)*10HO$ відділяється центрифугуванням, промивається 20 мл розбавленої щавлевої кислоти (6 г/л) і висушується при 1300 С; перетворення на оксид гадолінія досягається прожаренням при 8000 С. Подальше очищення може проводитися розчиненням в кислоті і осадженням у вигляді оксалата гадолінія. До рідини після центрифугування і промивань осаду (3300 мл) додають 2300 г КОН при інтенсивному перемішуванні до $ph = 12$, 6000 мл розчину з вмістом галію 55 г/л і гадолінія 1 г/л піддається електролізу при 600 Із з використанням катода з нержавіючої сталі і графітового анода при щільності струму близько 0,1 А/кв.см. Після 48 годин осідає 325 г

галію і залишається розчин з вмістом 0,4 г/л, що не піддається подальшій переробці. Обложений метал має чистоту 99,99 % і може бути безпосередньо перетворений в оксид.

При експлуатації ПК витрачаються наступні ресурси:

- електроенергія;
- папір для принтера;
- картріджи з фарбувальною стрічкою.

Для раціонального використання електроенергії не слід залишати включеним комп'ютер і принтер, якщо вони не потрібні в даний час. Друкувати можна з двох сторін. Витрати на папір навряд чи вдасться скоротити удвічі, проте економія буде вельми істотною. Проблему з утилізацією паперових відходів може вирішити вторинна переробка.

Для повторного використання картріджів з фарбувальною стрічкою необхідно купити пристрій для просочення стрічок і тоді картріджи можна буде використовувати 20-40 раз.

Сучасна технологія виготовлення елементів засобів обчислювальної техніки (СВТ) дозволяє досягти дуже низького рівня відмов елементів під час експлуатації. У зв'язку з цим відпадає необхідність проведення ремонтних робіт на місці експлуатації сучасних засобів обчислювальної техніки і як наслідок не утворюються відходи (несправні мікросхеми), що містять дорогоцінні і рідкоземельні метали. Природно, в сервісних центрах, що спеціалізуються на ремонті і технічному обслуговуванні СВТ, має бути організований збір і облік матеріалів, що містять коштовні метали, з подальшою обробкою цих матеріалів на спеціалізованих заводах з метою їх вилучення. У зв'язку з тим, що вітчизняне виробництво сучасних компонентів інформаційних технологій знаходиться в

сьогоднішні дні лише в зачатковому стані, СВТ складаються з парку імпортованих машин і устаткування. Із-за відсутності інформації про вміст дорогоцінних металів в елементах устаткування, строгий облік не представляється можливим і має бути покладений на фахівців експортних фірм. В умовах ринкової економіки підприємства мають бути самі зацікавлені у вторинній переробці, що містять дорогоцінні метали вузлів і елементів за умови неможливості їх використання.

Сумарна маса дорогоцінних металів в ПК, використовуваному в розробці даної дипломної роботи приблизно складає: золото - 0,23 г; срібло - 5,09 г.

Технологічний процес витягання дорогоцінних металів з друкарських плат здійснюється за наступною схемою. Друкарські плати сортуються по переважанню в них кількості дорогоцінних металів, дробляться і подрібнюються, обпалюються і плавляться. В процесі випалення перолітичеському розкладанню піддається пластмасова основа, а основа дорогоцінних металів у вигляді металевих залишків відновлюється до оксидів. Металевий залишок подрібнюється, гранулюється, проходить магнітну сепарацію і відбувається відділення магнітних від немагнітних часток. Отриманий таким чином порошок, розділений по видах дорогоцінних металів, у вигляді гранул розплавляється в індукційних плавильних печах з подальшим розділенням кожного металу окремо.

ВИСНОВКИ

В кваліфікаційній роботі вирішене наукове завдання з підвищення достовірності оцінювання метрики RFC застосунків з відкритим кодом на Java за допомогою нелінійної регресійної моделі.

1) Перевірку даних для побудови нелінійної регресійної моделі для оцінювання метрики RFC застосунків з відкритим кодом на Java на наявність викидів виконано за методом, який оснований на нормалізуючих перетвореннях і квадраті відстані Махаланобіса. З'ясовано, що за методом, який оснований на нормалізуючих перетвореннях і квадраті відстані Махаланобіса, тривимірні дані містять два багатовимірних викиди. Застосування критерія на основі квадрата відстані Махаланобіса до даних без їх нормалізації показує, що дані містять три багатовимірних викиди, які не можуть бути всі визнані викидами завдяки тому, що розподіл цих даних не є нормальним.

2) Удосконалено двофакторну нелінійну регресійну модель для оцінювання метрики RFC в залежності від метрик CBO та WMC застосунків з відкритим кодом на Java на основі нормалізуючого перетворення у формі десяткового логарифму із врахуванням викидів у регресії. Ця нелінійна регресійна модель у порівнянні з існуючими регресійними моделями, що побудовані без врахуванням викидів у регресії, має більші значення множинного коефіцієнту детермінації і відсотка прогнозування, менші значення середньої величини відносної похибки та ширин довірчого інтервалу і інтервалу прогнозування для більшої кількості значень метрик CBO та WMC. Зазначена нелінійна регресійна модель у порівнянні з лінійною має кращий показник $PRED(0,25)$ на 12,5%.

3) Отримані результати свідчать про те, що для побудови нелінійних регресійних моделей для оцінювання метрики RFC застосунків з відкритим кодом на Java потрібно використовувати нормалізуючі перетворення з урахуванням викидів у нелінійній регресії.

4) На підставі запропонованої нелінійної регресійної моделі розроблено програму для оцінювання метрики RFC застосунків з відкритим кодом на Java для системи моделювання Scilab 6.0.0. Була розроблена наступна програмна документація: технічне завдання, текст програми, опис програми, інструкція користувача, програма та методика випробувань, що наведені відповідно у додатках А, Б, В, Г, Д. Були проведені випробування програми згідно з програмою та методикою випробувань, що наведена у додатку Д. За результатами випробувань зроблено висновок про працездатність розробленої програми для оцінювання метрики RFC застосунків з відкритим кодом на Java для системи моделювання Scilab 6.0.0.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Laing V. Principal Components of Orthogonal Object-Oriented Metrics / V. Laing, C. Coleman // 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), 2001.
2. Chidamber & Kemerer object-oriented metrics suite. Access mode: <https://www.aivosto.com/project/help/pm-oo-ck.html>
3. Montgomery D.C. Introduction to Linear Regression Analysis / D. C. Montgomery, E. A. Peck, G. G. Vining // 6th ed., Wiley, 2021.
4. Bates D.M. Nonlinear Regression Analysis and Its Applications (2nd ed.) / D. M. Bates, D. G. Watts. New York: John Wiley & Sons, 1988. <http://dx.doi.org/10.1002/9780470316757>
5. Seber G.A.F. Nonlinear Regression / G. A. F. Seber, C. J. Wild. John Wiley & Sons, New York, 1989. <http://dx.doi.org/10.1002/0471725315>
6. Prykhodko N.V. Constructing the non-linear regression models on the basis of multivariate normalizing transformations / N. V. Prykhodko, S. B. Prykhodko // Electronic modeling. – 2018. – T.40. – № 6. – P. 99-108. – ISSN 0204–3572. DOI: <https://doi.org/10.15407/emodel.40.06.101>
7. Prykhodko S. Mathematical Modeling of Non-Gaussian Dependent Random Variables by Nonlinear Regression Models Based on the Multivariate Normalizing Transformations / S. Prykhodko, N. Prykhodko // In: Shkarlet S., Morozov A., Palagin A. (eds) Mathematical Modeling and Simulation of Systems (MODS'2020). MODS 2020. Advances in Intelligent Systems and Computing,

vol. 1265, 2021. – P. 166-174. Springer, Cham. DOI: https://doi.org/10.1007/978-3-030-58124-4_16

8. Prykhodko S. A Modified Technique for Constructing Nonlinear Regression Models Based on the Multivariate Normalizing Transformations / Sergiy Prykhodko, Natalia Prykhodko // 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. – 2022. – Vol.3179. – CEUR-WS.org – P. 156-166. ISSN 1613-0073. (Published on CEUR-WS: 2-Aug-2022) Access mode: http://ceur-ws.org/Vol-3179/Paper_15.pdf

9. Приходько С.Б. Нелінійна регресійна модель для оцінювання метрики RFC застосунків з відкритим кодом на Java / С. Б. Приходько, **Р. О. Позняков** // Інформаційні технології: моделі, алгоритми, системи (ITMAS – 2022): Матеріали III-ої Всеукраїнської науково-практичної інтернет конференції (26-28 жовтня 2022 р.). – Миколаїв: НУК імені адмірала Макарова, 2022. – С. 8-9. Режим доступу: <http://itconf.nuos.edu.ua/2022/wp-content/uploads/sites/5/NUOS-ITMAS-2022-Proceedings.pdf>

10. Mardia K.V. Measures of multivariate skewness and kurtosis with applications / K. V. Mardia // Biometrika. – 1970. – Vol. 57. – P. 519–530. DOI: <https://doi.org/10.1093/biomet/57.3.519>

11. Johnson R.A. Applied Multivariate Statistical Analysis / R. A. Johnson, D. W. Wichern. – Pearson Prentice Hall, 2007. – 800 p.

12. Prykhodko S. Detecting Outliers in Multivariate Non-Gaussian Data on the basis of Normalizing Transformations / S. Prykhodko, N. Prykhodko, L. Makarova, K. Pugachenko // Proceedings of the 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON) «Celebrating 25 Years of IEEE

Ukraine Section», Kyiv, Ukraine, May 29 – June 2, 2017. – P. 846-849. DOI: <https://doi.org/10.1109/UKRCON.2017.8100366>

13. Prykhodko S. Application of the Squared Mahalanobis Distance for Detecting Outliers in Multivariate Non-Gaussian Data / S. Prykhodko, N. Prykhodko, L. Makarova, A. Pukhalevych // Proceedings of 14th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET), Lviv-Slavske, Ukraine, February 20–24, 2018. – P. 962-965. DOI: <https://doi.org/10.1109/TCSET.2018.8336353>

14. Foss T. A simulation study of the model evaluation criterion MMRE / T. Foss, E. Stensrud, B. Kitchenham, I. Myrtveit // IEEE Transactions on software engineering. – 2003. – 11(29). – P. 985–995. DOI: <https://doi.org/10.1109/TSE.2003.1245300>

15. Port, D. Comparative studies of the model evaluation criterions MMRE and PRED in software cost estimation research / D. Port, M. Korte // In: Proceedings of the 2nd ACM-IEEE International Symposium on Empirical Software Engineering and Measurement. – ACM, New York, 2008. – P. 51–60. DOI: <https://doi.org/10.1145/1414004.1414015>

16. Prykhodko S. Outlier Detection in Non-Linear Regression Analysis Based on the Normalizing Transformations / Sergiy Prykhodko, Natalia Prykhodko, Lidiia Makarova, Andrii Pukhalevych // Proceedings of the 15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET), IEEE, Lviv-Slavske, 2020, P. 407-410. <https://doi.org/10.1109/TCSET49122.2020.235464>

17. 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

18. Закон України «Про охорону праці» (із змінами). Режим доступу: <https://zakon.rada.gov.ua/laws/show/2694-12#Text>

19. Законом України № 1268-ХІІ від 26.06.91 р. «Про охорону навколишнього природного середовища» (із змінами). Режим доступу: <https://zakon.rada.gov.ua/laws/show/1264-12#Text>

Додаток А – Технічне завдання на розробку програми для оцінювання метрики RFC застосунків з відкритим кодом на Java

Вступ

Назва розробки – “Програма для оцінювання метрики RFC застосунків з відкритим кодом на Java”, далі програма.

Підстави для розробки

Завдання на кваліфікаційну роботу.

Призначення розробки:

Програма призначена для оцінювання метрики RFC застосунків з відкритим кодом на Java за нелінійним рівнянням регресії.

Вимоги до програми:

Вимоги функціональних характеристик.

Вимоги до складу функцій, що виконуються.

Програма має виконувати такі функції:

- 1) Вводити з текстового файлу емпіричні дані з метрик RFC, CBO та WMC застосунків з відкритим кодом на Java.
- 2) Вводити з інтерфейсу програми довірчу ймовірність, дві метрики CBO та WMC застосунку, для якого потрібно здійснити оцінювання метрики RFC.
- 3) Виконувати нормалізацію емпіричних даних за нормалізуючим перетворенням у формі десяткового логарифму.
- 4) Знаходити значення коефіцієнтів лінійного рівняння регресії для нормалізованих емпіричних даних.
- 5) Здійснювати побудову нелінійної регресійної моделі.

6) Обчислювати значення коефіцієнту детермінації R^2 та величини відносної похибки MRE.

7) Здійснювати побудову інтервалу прогнозування та довірчого інтервалу нелінійної регресії.

8) Виводити на екран нелінійну регресію, довірчий інтервал та інтервал прогнозування нелінійної регресії метрики RFC, значення коефіцієнту детермінації R^2 та величини відносної похибки MRE.

9) Виводити у текстовий файл результати оцінювання метрики RFC застосунку з відкритим кодом на Java.

Вимоги до організації вхідних та вихідних даних

Вхідні дані:

- емпіричні дані з метрик RFC, CBO та WMC застосунків з відкритим кодом на Java;

- значення довірчої ймовірності, метрик CBO та WMC застосунку, для якого потрібно здійснити оцінювання метрики RFC.

Вихідні дані:

- значення нелінійної регресії, границь довірчого інтервалу та інтервалу прогнозування нелінійної регресії метрики RFC застосунку.

- значення коефіцієнту детермінації та відносної похибки MRE.

Вимоги до надійності не висуваються.

Вимоги до умов експлуатації не висуваються.

Вимоги до складу й параметрів технічних засобів.

Програма має експлуатуватись на обладнанні з параметрами не нижчими за наступні: процесор Pentium IV (або еквівалентний) з інструкціями SSE2,

оперативна пам'ять 2 Гб (1 Гб мінімум), простір на жорсткому диску 600 Мб мінімум.

Вимоги до інформаційної та програмної сумісності.

Для запуску та роботи з програмним забезпеченням необхідна операційна система Windows Vista/7/8/10 (32 або 64 біти), система моделювання Scilab 6.0.0., компілятор C (Visual Studio 2010 або Visual Express 2010) для компіляції зовнішніх модулів C або C++ та використання моделі в Xcos.

Вимоги до програмної документації.

Програмна документація має включати: технічне завдання на розробку програми, текст програми, опис програми, інструкція користувача, програма та методика випробувань.

Техніко-економічні показники не розраховуються.

Стадії та етапи розробки.

Стадії та етапи розробки програми представлені у таблиці.

Таблиця – Стадії та етапи розробки

Стадія розробки	Етапи робіт	Термін виконання
1. Технічне завдання	Розробка та затвердження технічного завдання	11.10.2022
2. Ескізний проект	Розробка ескізного проекту	13.10.2022
	Затвердження ескізного проекту	14.10.2022
3. Технічний проект	Розробка технічного проекту	20.10.2022
	Затвердження технічного проекту	21.10.2022
4. Робочий проект	Розробка програми	04.11.2022
	Розробка програмної документації	10.11.2022
	Випробування програми	11.11.2022

Порядок контролю і приймання

Порядок контролю та приймання програми здійснюється представниками замовника у присутності представника розробника.

Для приймання програми надається програмна документація, яка була розроблена відповідно до технічного завдання. Представники замовника установлюють відповідність програми технічному завданню.

За результатами приймання програми представниками замовника складається акт приймання програми.

Додаток Б – Текст програми

```
//Оцінювання метрики RFC застосунків з відкритим кодом на Java
f=figure();//Создание графического окна.
//Устанавливаем размеры окна.
set(f,'position',[0,0,700,300])
//Устанавливаем заголовок окна.
set(f,'figure_name',...
'Оцінювання метрики RFC застосунків з відкритим кодом на Java');

lab_text0=uicontrol(f,'style','text','string',...
'Файл з даними','position',...
[350,270,80,20]);
edit_text=uicontrol(f,'style','edit','string',...
'D:\Data_F\RFC_CBO_WMC46.txt','position',[320,240,140,20]);

BtDatInp=uicontrol('style','pushbutton','string','Ввести',...
'CallBack','[x,yx,f1,f2,n]=DataInput()','position',[340,210,100,20]);

lab_text2=uicontrol(f,'style','text','string',...
'Наявність фактичної метрики RFC','position',...
[500,270,190,20]);
lab_text3=uicontrol(f,'style','text','string',...
'Дані для оцінювання','position',...
[60,270,110,20]);
lab_text4=uicontrol(f,'style','text','string',...
'Значення СВО на рівні застосунку','position',...
[20,240,200,20]);
edit_text4=uicontrol(f,'style','edit','string',...
'8.625','position',[230,240,60,20]);
lab_text5=uicontrol(f,'style','text','string',...
'Значення WMC на рівні застосунку','position',...
[20,210,200,20]);
edit_text5=uicontrol(f,'style','edit','string',...
'22.25','position',[230,210,60,20]);
lab_text6=uicontrol(f,'style','text','string',...
'Довірча ймовірність','position',...
[20,180,200,20]);
edit_text6=uicontrol(f,'style','edit','string',...
'0.95','position',[230,180,60,20]);
lab_text8=uicontrol(f,'style','text','string',...
'Результати оцінювання','position',...
[50,120,120,20]);
lab_text9=uicontrol(f,'style','text','string',...
'Результат нел. регресії метрики RFC','position',...
[20,90,245,20]);
lab_text10=uicontrol(f,'style','text','string',...
'Нижня границя довір.інтервалу метрики RFC','position',...
[20,60,245,20]);
lab_text11=uicontrol(f,'style','text','string',...
'Верхня границя довір.інтервалу метрики RFC','position',...
[20,30,245,20]);
```

```

lab_text12=uicontrol(f,'style','text','string',...
'Нижня границя інтервалу прогноз. метрики RFC','position',...
[355,60,260,20]);
lab_text13=uicontrol(f,'style','text','string',...
'Верхня границя інтервалу прогноз. метрики RFC','position',...
[355,30,260,20]);
//
//Текстовое поле, определяющее вывод результатов.
textresult=uicontrol(f,'style','text','string','', 'position',...
[270,90,70,20]);
textresult2=uicontrol(f,'style','text','string','', 'position',...
[270,60,70,20]);
textresult3=uicontrol(f,'style','text','string','', 'position',...
[270,30,70,20]);
textresult4=uicontrol(f,'style','text','string','', 'position',...
[190,120,250,20]);
textresult5=uicontrol(f,'style','text','string','', 'position',...
[620,60,70,20]);
textresult6=uicontrol(f,'style','text','string','', 'position',...
[620,30,70,20]);

//Флажок, отвечающий за выбор.
radio_bikv_x=uicontrol('style','radiobutton','string',...
'е (включити), немає (виключити)','value',1,'position',...
[490,240,200,20]);
//
//Проверяем значение флажка, если флажок выключен,
if (get(radio_bikv_x,'value')==0)

else
// 'Нелінійне рівняння регресії не знайдено, моделювання не виконано');
edit_text12=uicontrol(f,'style','edit','string',..
'0','position',[500,210,60,20]);
end;
Btnr_pred_int=uicontrol('style','pushbutton','string','Оцінювання','CallBack',...
.
'RFC_AppEstim(x,yx,n)', 'position',[340,150,100,20]);

BtClose=uicontrol('style','pushbutton','string','Закрити',...
'CallBack','W_Close()', 'position',[350,5,100,20]);

function [x, yx, f1, f2, n]=DataInput()
dir_name=get(edit_text,'string');

file_PC=mopen(dir_name,'r');
file_PC2=mopen('D:\Data_F\Web_Django_30_data2.txt','w+');
file_PC3=mopen('D:\Data_F\Web_Django_30_data3.txt','w+');
i=1;
//n=52;
while (feof(file_PC))<1
    y(i)=mfscanf(file_PC,'%f'); x1(i)=mfscanf(file_PC,'%f');
    x2(i)=mfscanf(file_PC,'%f');
    i=i+1;
end;

```

```

n=i-1;

fclose(file_PC);

for i = 1:n
    yx(i,1)=y(i); yx(i,2)=x1(i); yx(i,3)=x2(i);
    x(i,1)=x1(i); x(i,2)=x2(i);
end;
fprintf(file_PC2, '%f\t%f\t%f\n', y, x1, x2);

for i = 1:n
    for j = 1:2 fprintf(file_PC3, "%5.3f ", x(i,j)); end;
    fprintf(file_PC3, "\n");
end;

fclose(file_PC2);
fclose(file_PC3);
f1=figure(); //Создание графического окна.
//Устанавливаем размеры окна.
set(f1, 'position', [50,50,700,300]);
//Устанавливаем заголовок окна.
set(f1, 'figure_name', 'Емпіричні дані: RFC в залежності від СВО');
plot2d(x1,y,-9);

f2=figure(); //Создание графического окна.
//Устанавливаем размеры окна.
set(f2, 'position', [50,50,700,300]);
//Устанавливаем заголовок окна.
set(f2, 'figure_name', 'Емпіричні дані: RFC в залежності від WMC');
plot2d(x2,y,-9);

x1A=0; x2A=0;
for i=1:n
    x1A=x1A+x(i,1); x2A=x2A+x(i,2);
end;
x1A=x1A/n; x2A=x2A/n;
Sx1x1=0; Sx2x2=0;
Sx1x2=0;
for i=1:n
    Sx1x1=Sx1x1+(x(i,1)-x1A)*(x(i,1)-x1A);
    Sx1x2=Sx1x2+(x(i,1)-x1A)*(x(i,2)-x2A);
    Sx2x2=Sx2x2+(x(i,2)-x2A)*(x(i,2)-x2A);
end;
Sxx(1,1)=Sx1x1; Sxx(1,2)=Sx1x2;
Sxx(2,1)=Sx1x2; Sxx(2,2)=Sx2x2;

Sxx_1=1/Sxx;

z1A=0; z2A=0;
for i=1:n
    z1A=z1A+log10(x(i,1)); z2A=z2A+log10(x(i,2));
end;
z1A=z1A/n; z2A=z2A/n;
Sz1z1=0; Sz2z2=0;

```

```

Sz1z2=0;
for i=1:n
    Sz1z1=Sz1z1+(log10(x(i,1))-z1A)*(log10(x(i,1))-z1A);
    Sz1z2=Sz1z2+(log10(x(i,1))-z1A)*(log10(x(i,2))-z2A);
    Sz2z2=Sz2z2+(log10(x(i,2))-z2A)*(log10(x(i,2))-z2A);
end;
Szz(1,1)=Sz1z1; Szz(1,2)=Sz1z2;
Szz(2,1)=Sz1z2; Szz(2,2)=Sz2z2;

Szz_1=1/Szz;

zi(1)=log10(x(1,1))-z1A; zi(2)=log10(x(1,2))-z2A;
ziT=zi';
ziSzz_1=ziT*Szz_1;
result=ziSzz_1*zi;

Df=n-3;
P=0.975;
Q=1-P;

[T]=cdfst("T",Df,P,Q);

b0=-0.05477;
b1=0.67226;
b2=0.44154;

Szyzyn4=0;
for i=1:n
    Szyzyn4=Szyzyn4+...
        (b0+b1*log10(x(i,1))+b2*log10(x(i,2))-log10(yx(i,1)))^2;
end;
Szyzyn4=Szyzyn4/(n-4)
Delta_z=T*(Szyzyn4*(1/n+result))^0.5;

zy_result=b0+b1*log10(x(1,1))+b2*log10(x(1,2));

zyLBC=zy_result-Delta_z; zyUBC=zy_result+Delta_z;

y_result=10^zy_result;
yLBC=10^zyLBC; yUBC=10^zyUBC;

file_PC5=fopen('D:\Data_F\Web_Django_30_data5.txt','w+');
//for i=1:3
    fprintf(file_PC5, "%5.6f ", yUBC);
//end

fclose(file_PC5);

file_PC4=fopen('D:\Data_F\Web_Django_30_data4.txt','w+');
for i = 1:2
    for j = 1:2 fprintf(file_PC4, "%5.6f ", Szz_1(i,j)); end;
    fprintf(file_PC4, "\n");
end;
fclose(file_PC4);

```

endfunction

```
//Функція оцінювання метрики RFC застосунку з відкритим кодом на Java
function RFC AppEstim(x, yx, n)
//Считываем значение переменных из текстовых полей и
//преобразовываем их к числовому типу.
x1i=eval(get(edit_text4,'string'));
x2i=eval(get(edit_text5,'string'));
Pconf=eval(get(edit_text6,'string'));

x1A=0; x2A=0;
yA=0;
for i=1:n
    x1A=x1A+x(i,1); x2A=x2A+x(i,2);
    yA=yA+yx(i,1);
end;
x1A=x1A/n; x2A=x2A/n;
yA=yA/n;
Sx1x1=0; Sx2x2=0;
Sx1x2=0;
SyA=0;
for i=1:n
    Sx1x1=Sx1x1+(x(i,1)-x1A)*(x(i,1)-x1A);
    Sx1x2=Sx1x2+(x(i,1)-x1A)*(x(i,2)-x2A);
    Sx2x2=Sx2x2+(x(i,2)-x2A)*(x(i,2)-x2A);
    SyA=SyA+(yx(i,1)-yA)*(yx(i,1)-yA);
end;
Sxx(1,1)=Sx1x1; Sxx(1,2)=Sx1x2;
Sxx(2,1)=Sx1x2; Sxx(2,2)=Sx2x2;

Sxx_1=1/Sxx;

z1A=0; z2A=0;
//z3A=0;
for i=1:n
    z1A=z1A+log10(x(i,1)); z2A=z2A+log10(x(i,2));
end;
z1A=z1A/n; z2A=z2A/n;
Sz1z1=0; Sz2z2=0;
Sz1z2=0;
for i=1:n
    Sz1z1=Sz1z1+(log10(x(i,1))-z1A)*(log10(x(i,1))-z1A);
    Sz1z2=Sz1z2+(log10(x(i,1))-z1A)*(log10(x(i,2))-z2A);
    Sz2z2=Sz2z2+(log10(x(i,2))-z2A)*(log10(x(i,2))-z2A);
end;
Szz(1,1)=Sz1z1; Szz(1,2)=Sz1z2;
Szz(2,1)=Sz1z2; Szz(2,2)=Sz2z2;

Szz_1=1/Szz;

zi(1)=log10(x1i)-z1A; zi(2)=log10(x2i)-z2A;
ziT=zi';
ziSzz_1=ziT*Szz_1;
```

```

result=ziSzz_1*zi;

Df=n-3;
alpha=1-Pconf;
P=1-alpha/2;
//P=0.975;
Q=1-P;

[T]=cdf('T',Df,P,Q);

b0=-0.05477;
b1=0.67226;
b2=0.44154;

Szyzyn4=0; Syy=0;
for i=1:n
    Szyzyn4=Szyzyn4+...
        (b0+b1*log10(x(i,1))+b2*log10(x(i,2))-log10(yx(i,1)))^2;
    Syy=Syy+(yx(i,1)-10^b0*x(i,1)^b1*x(i,2)^b2)^2;
end;
Szyzyn4=Szyzyn4/(n-3)
Delta_z=T*(Szyzyn4*(1/n+result))^0.5;
Delta_z_pr=T*(Szyzyn4*(1+1/n+result))^0.5;

r_2=1-Syy/SyA;

zy_result=b0+b1*log10(x1i)+b2*log10(x2i);

zyLBC=zy_result-Delta_z; zyUBC=zy_result+Delta_z;

zyLBPr=zy_result-Delta_z_pr; zyUBPr=zy_result+Delta_z_pr;

y_result=10^zy_result;
yLBC=10^zyLBC; yUBC=10^zyUBC;
yLBPr=10^zyLBPr; yUBPr=10^zyUBPr;

file_PC5=fopen('D:\Data_F\Web_Django_30_data5.txt','w+');
//for i=1:3
    fprintf(file_PC5, "%5.6f ", yUBC);
//end
fclose(file_PC5);

file_PC4=fopen('D:\Data_F\Web_Django_30_data4.txt','w+');
for i = 1:2
    for j = 1:2 fprintf(file_PC4, "%5.6f ", Szz_1(i,j)); end;
    fprintf(file_PC4, "\n");
end;
fclose(file_PC4);

set(textresult,'string', sprintf("=%1.4f",y_result));
set(textresult2,'string', sprintf("=%1.4f",yLBC));
set(textresult3,'string', sprintf("=%1.4f",yUBC));
set(textresult4,'string',...
    sprintf("Коефіцієнт детермінації R^2=%1.4f",r_2));

```



```

set(textresult5,'string', sprintf("=%1.4f",yLBPr));
set(textresult6,'string', sprintf("=%1.4f",yUBPr));

if (get radio_bikv_x,'value')==0)

else
    yfact=eval(get(edit_text12,'string'));
    MRE=abs((yfact-y_result)/yfact);
    textresult5=uicontrol(f,'style','text','string','','position',...
    [500,120,160,20]);
    set(textresult5,'string',...
    sprintf("Відносна похибка MRE=%1.4f",MRE));
end;

endfunction

//Функция закрытия окна.
function W_Close()
close(f);
close(f1);
close(f2);
endfunction

```

ДОДАТОК В – Опис програми

Загальні відомості

Програма для оцінювання метрики RFC застосунків з відкритим кодом на Java, для системи моделювання Scilab 6.0.0. Файл програми RFC_Estim_LG.sce має розмір 14 КБ.

Функціональне призначення

Програма RFC_Estim_LG призначений для оцінювання метрики RFC застосунків з відкритим кодом на Java, для системи моделювання Scilab 6.0.0.

Технічні засоби, що використовуються

Програма має експлуатуватись на обладнанні з параметрами не нижчими за наступні: процесор Pentium IV (або еквівалентний) з інструкціями SSE2, оперативна пам'ять 2 Гб (1 Гб мінімум), простір на жорсткому диску 600 Мб мінімум.

Виклик та завантаження

Для запуску та роботи з програмою необхідна операційна система Windows Vista/7/8/10 (32 або 64 біти), система моделювання Scilab 6.0.0.

Завантаження програми здійснюється шляхом відкриття файлу RFC_Estim_LG.sce стандартними засобами ОС.

ДОДАТОК Г – Інструкція користувача

Загальний опис роботи з програмою

Після завантаження файлу RFC_Estim_LG.sce на екрані з'являється форма для відображення початкового стану програми для оцінювання метрики RFC застосунків з відкритим кодом на Java (рисунок Г.1).

The screenshot shows a Windows-style application window titled "Оцінювання метрики RFC застосунків з відкритим кодом на Java". The window has a menu bar with "File", "Tools", "Edit", and "?". Below the menu bar is a toolbar with icons for file operations. The main area is divided into several sections:

- Дані для оцінювання**: Contains three input fields:
 - Значення CBO на рівні застосунку: 8.625
 - Значення WMC на рівні застосунку: 22.25
 - Довірча ймовірність: 0.95
- Файл з даними**: Contains a text field with the value "ta_F\RFC_CBO_WMC46.txt" and a "Ввести" button.
- Наявність фактичної метрики RFC**: Contains a radio button labeled "є (включити), немає (виключити)" which is selected, and a numeric input field with the value "0".
- Оцінювання**: A button located below the "Ввести" button.
- Результати оцінювання**: A section with several output fields:
 - Результат нел. регресії метрики RFC
 - Нижня границя довір. інтервалу метрики RFC
 - Верхня границя довір. інтервалу метрики RFC
 - Нижня границя інтервалу прогноз. метрики RFC
 - Верхня границя інтервалу прогноз. метрики RFC
- Закрити**: A button at the bottom center of the window.

Рисунок Г.1 – Форма для відображення початкового стану програми

Далі у вікні редагування «Файл з даними» графічного вікна «Оцінювання метрики RFC застосунків з відкритим кодом на Java» (рис.Г.1) потрібно відредагувати ім'я текстового файлу з початковими емпіричними даними, що

будуть використовуватися для оцінювання метрики RFC застосунків з відкритим кодом на Java.

Після натискання на командну кнопку «Ввести» з текстового файлу вводяться дані.

Потім у графічному вікні «Оцінювання метрики RFC застосунків з відкритим кодом на Java» (рис. Г.1) у відповідних вікнах редагування потрібно відредагувати: значення метрик СВО та WMC застосунку, довірчу ймовірність.

Крім того у графічному вікні «Оцінювання метрики RFC застосунків з відкритим кодом на Java» (рис. Г.1) потрібно за допомогою радіо кнопки здійснити вибір наявності фактичного значення метрики RFC застосунку та ввести його для подальшого порівняння з відповідною оцінкою метрики RFC застосунку для введених значень метрик СВО та WMC.

Далі потрібно натиснути на командну кнопку «Оцінювання». Після цього з'являться результати оцінювання метрики RFC застосунку для введених значень метрик СВО та WMC.

І на останок у графічному вікні «Оцінювання метрики RFC застосунків з відкритим кодом на Java» головного модуля програми (рис. Г.1) потрібно натиснути на командну кнопку «Закрити». Це приведе до закриття всіх графічних вікон та завершення програми.

ДОДАТОК Д – Програма та методика випробувань

Об’єкт випробувань: Програма для оцінювання метрики RFC застосунків з відкритим кодом на Java, RFC_Estim_LG.

Ціль випробувань: перевірка відповідності програми RFC_Estim_LG до вимог технічного завдання і повноти реалізації.

Склад програмної документації:

- технічне завдання;
- інструкція користувача;
- опис програми;
- текст програми;
- програма та методика випробувань.

Технічні вимоги.

Програма має експлуатуватись на обладнанні з параметрами не нижчими за наступні: процесор Pentium IV (або еквівалентний) з інструкціями SSE2, оперативна пам’ять 2 Гб (1 Гб мінімум), простір на жорсткому диску 600 Мб мінімум.

Методи випробувань.

Випробування програми виконується представником замовника в присутності представника розробника. Відповідно до опису програми виконується запуск програми RFC_Estim_LG. Далі виконується перевірка роботи програми. Виконується перевірка всіх екранних форм програми, а також всіх допустимих для виконання операцій всередині екранних форм згідно до посібника користувача. Якщо програма демонструє адекватне функціонування згідно до вимог ТЗ та представленої програмної документації, то робота з

розробки програми вважається завершеною, і виконується впровадження програми до експлуатації. В протилежному випадку програма має бути відправлено на доробку розробникові.

Використовуються тести за методологією „чорної скриньки” (рис.Д.1-Д.5).

Оцінювання метрики RFC застосунків з відкритим кодом на Java

File Tools Edit ?

Оцінювання метрики RFC застосунків з відкритим кодом на Java

Дані для оцінювання		Файл з даними	Наявність фактичної метрики RFC	
Значення CBO на рівні застосунку	8.625	ta_F\RFC_CBO_WMC46.txt	☒ є (включити), немає (виключити)	
Значення WMC на рівні застосунку	22.25	Ввести	0	
Довірча ймовірність	0.95			
		Оцінювання		
Результати оцінювання				
Результат нел. регресії метрики RFC				
Нижня границя довір.інтервалу метрики RFC		Нижня границя інтервалу прогноз. метрики RFC		
Верхня границя довір.інтервалу метрики RFC		Верхня границя інтервалу прогноз. метрики RFC		
Закрити				

Рисунок Д.1 – Графічне вікно головного модуля програми

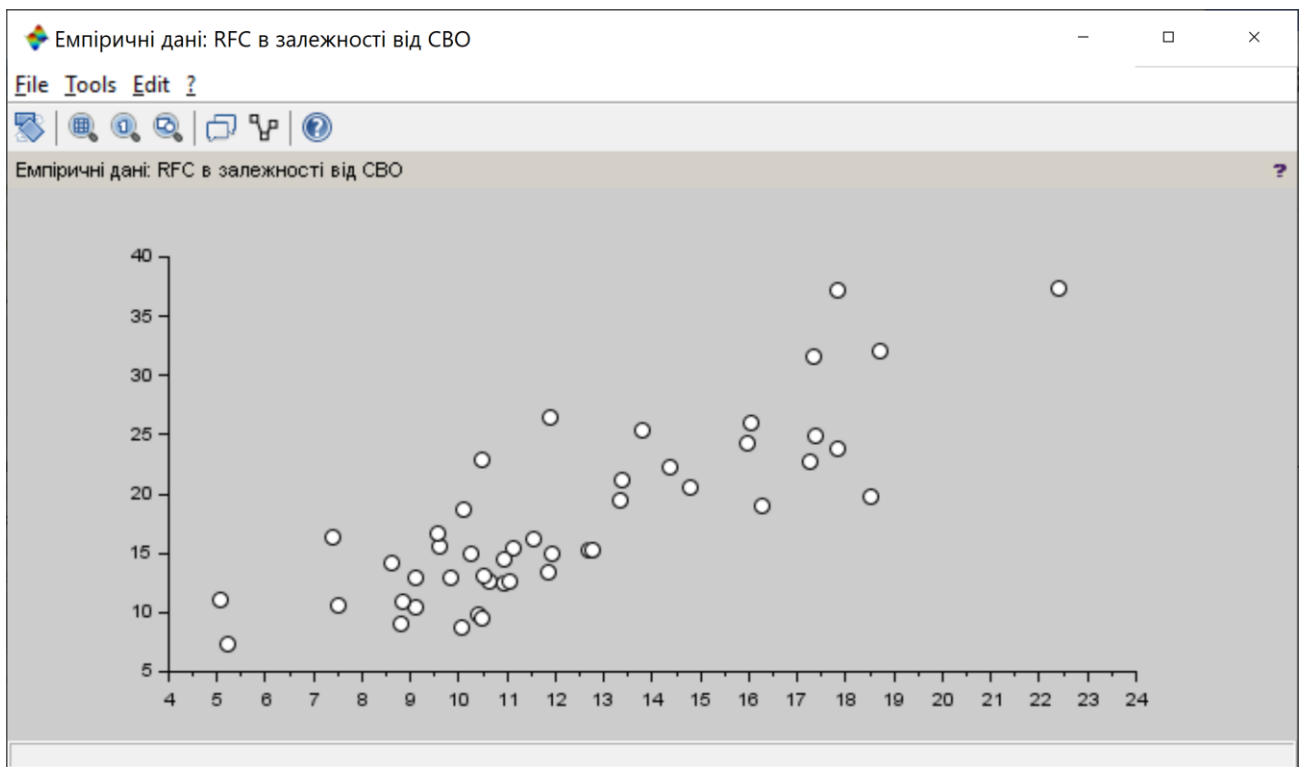


Рисунок Д.2 – Візуалізація тестових емпіричних даних.

Результати роботи програми після натискання на командну кнопку «Оцінювання» у разі, коли радіо кнопка виключена, наведені на рисунку Д.3.

Оцінювання метрики RFC застосунків з відкритим кодом на Java

File Tools Edit ?

Оцінювання метрики RFC застосунків з відкритим кодом на Java

Дані для оцінювання		Файл з даними	Наявність фактичної метрики RFC	
Значення CBO на рівні застосунку	8.625	ta_F\RFC_CBO_WMC46.txt	☐ є (включити), немає (виключити)	
Значення WMC на рівні застосунку	22.25	Ввести	0	
Довірча ймовірність	0.95			
Оцінювання				
Результати оцінювання		Коефіцієнт детермінації $R^2=0.8718$		
Результат нел. регресії метрики RFC	=14.7640			
Нижня границя довір.інтервалу метрики RFC	=13.6332	Нижня границя інтервалу прогноз. метрики RFC	=10.7140	
Верхня границя довір.інтервалу метрики RFC	=15.9886	Верхня границя інтервалу прогноз. метрики RFC	=20.3450	
Закрити				

Рисунок Д.3 – Результати роботи програми після натискання на командну кнопку «Оцінювання» у разі, коли радіо кнопка виключена

Результати роботи програми після натискання на командну кнопку «Оцінювання» у разі, коли радіо кнопка включена та введено фактичне значення метрики RFC – 14,125, наведені на рисунку Д.4.

Оцінювання метрики RFC застосунків з відкритим кодом на Java

File Tools Edit ?

Оцінювання метрики RFC застосунків з відкритим кодом на Java

Дані для оцінювання		Файл з даними	Наявність фактичної метрики RFC	
Значення CBO на рівні застосунку	8.625	ta_F\RFC_CBO_WMC46.txt	☒ є (включити), немає (виключити)	
Значення WMC на рівні застосунку	22.25	Ввести	14.125	
Довірча ймовірність	0.95			
Оцінювання				
Результати оцінювання		Коефіцієнт детермінації $R^2=0.8718$		Відносна похибка $MRE=0.0452$
Результат нел. регресії метрики RFC		=14.7640		
Нижня границя довір.інтервалу метрики RFC		Нижня границя інтервалу прогноз. метрики RFC		=10.7140
Верхня границя довір.інтервалу метрики RFC		Верхня границя інтервалу прогноз. метрики RFC		=20.3450
Закрити				

Рисунок Д.4 – Результати роботи програми після натискання на командну кнопку «Оцінювання» у разі, коли радіо кнопка включена та введено фактичне значення метрики RFC – 14,125

При випробуванні програми потрібно ввести наступні тестові дані із текстового файлу:

```

9.8636 10.4091 10.2273
26.4477 11.9038 71.8368
8.6879 10.047 11.1007
15.3894 11.1295 17.634
23.8848 17.8436 28.4542
10.4876 9.1124 11.303
31.5262 17.3327 33.6835
11.1429 5.0714 14.6429
37.2778 22.4167 35.6944

```

7.3064 5.2047 12.9848
21.1303 13.3763 15.988
32.0096 18.7195 23.4093
14.5556 10.9242 17.0202
13.4284 11.8396 11.4272
10.8677 8.8377 10.7936
12.4876 10.9353 17.1542
19.4667 13.3333 11.5333
22.9474 10.4737 29.7368
9.0769 8.8077 11.3077
16.3509 7.3918 24.0936
22.2145 14.3478 17.9217
12.9688 9.832 17.1523
15.0196 10.2448 14.6171
19.7506 18.5265 18.994
14.125 8.625 22.25
24.2139 15.9626 25.2366
14.9219 11.9123 18.5455
12.6526 11.0559 14.7221
9.521 10.4902 8.042
25.9306 16.0278 25.0139
16.2646 11.5291 11.8027
18.6289 10.0773 15.5928
37.1579 17.8138 65.8381
12.9736 9.1031 20.253
12.5962 10.6263 14.0632
24.9355 17.3871 29.2839
15.5856 9.5901 15.5541
25.3875 13.8 42.1125
18.9659 16.2689 23.7955
16.7353 9.5588 15.3971
20.5263 14.7924 18.0322
10.6218 7.5192 12.9487
13.1 10.5286 13.7429
15.2406 12.6934 11.2736
15.3285 12.744 11.3816
22.7068 17.2398 25.2156