

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

«__» _____ 2021 р.

КВАЛІФІКАЦІЙНА РОБОТА
на здобуття ступеня вищої освіти «магістр»

на тему: Удосконалення нелінійної регресійної моделі для оцінювання розміру програмного забезпечення інформаційних систем з кодом на РНР та розробка програми для її реалізації

Виконав: студент 6151м групи
_____ Сербулов Г.В.
(підпис, ПІБ)

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

Миколаїв – 2021 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)

«___» _____ 2021 р.

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

Студенту Сербулову Глібу Васильовичу _____
 (Прізвище, ім'я, по батькові)

1. Тема роботи: Удосконалення нелінійної регресійної моделі для оцінювання розміру програмного забезпечення інформаційних систем з кодом на PHP та розробка програми для її реалізації _____

Керівник роботи Приходько С.Б. _____

Затверджені наказом ректора № 1239уч від «13» _____ 10 _____ 2021 р.

2. Термін подання роботи: 06.12.2021 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 _____

РЕФЕРАТ

Сербулов Гліб Васильович

«Удосконалення нелінійної регресійної моделі для оцінювання розміру програмного забезпечення ІС з кодом на РНР 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, 2021 р.

Обсяг роботи: 112 стор., 8 табл., 10 рис., 29 використаних джерела, 5 додатків.

Актуальність теми роботи: необхідність підвищення достовірності оцінювання розміру програмного забезпечення (ПЗ) інформаційних систем (ІС) з кодом на РНР.

Мета та завдання дослідження. Метою є підвищення достовірності оцінювання розміру ПЗ ІС з кодом на РНР за допомогою нелінійної регресійної моделі. Завдання дослідження: проаналізувати існуючі регресійні моделі для оцінювання розміру ПЗ ІС з кодом на РНР; удосконалити регресійну модель для оцінювання розміру ПЗ ІС з кодом на РНР; розробити програму для оцінювання розміру ПЗ ІС з кодом на РНР.

Об'єктом дослідження є процес оцінювання розміру ПЗ ІС з кодом на РНР. **Предметом дослідження** є регресійні моделі для оцінювання розміру ПЗ ІС з кодом на РНР.

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

Наукова новизна одержаних результатів. Удосконалено регресійну модель для оцінювання розміру ПЗ ІС з кодом на РНР на основі нормалізуючого перетворення Джонсона сімейства S_B , яка дозволяє підвищити достовірність оцінювання розміру ПЗ ІС з кодом на РНР в порівнянні з існуючими регресійними моделями.

Практичне значення одержаних результатів. Розроблено програму для оцінювання розміру ПЗ ІС з кодом на РНР 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 розміру програмного забезпечення, інформаційна система, РНР.

ABSTRACT

Serbulov Gleb Vasilyevich

«Improving the nonlinear regression model to estimate a software size of PHP-based information systems 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, 2021.

The qualification work is presented on the 112 pages of typewritten text, contains 8 tables, 10 figures, 5 appendices and 29 references.

Relevance of the topic of the work: the need to increase the confidence of estimating a software size of PHP-based information systems.

The goal and objectives of the study. The goal of the study is to increase the confidence of estimating a software size of PHP-based information systems by nonlinear regression model. The objectives of the study: to analyze existing regression models to estimate a software size of PHP-based information systems; to improve the regression model for estimating a software size of PHP-based information systems; to develop the software for estimating a software size of PHP-based information systems.

The object of the study is the process of estimating a software size of PHP-based information systems.

The subject of the study is the regression models for estimating a software size of PHP-based information systems.

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 a software size of PHP-based information systems is improved based on the Johnson normalizing transformation of S_B family which allows to increase confidence of estimating a software size of PHP-based information systems in comparison with existing nonlinear 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, software size estimation, information system, PHP.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ	8
ВСТУП	9
1 АНАЛІЗ ІСНУЮЧИХ МОДЕЛЕЙ ДЛЯ ОЦІНЮВАННЯ РОЗМІРУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ІС З КОДОМ НА РНР ТА ОБҐРУНТУВАННЯ НЕОБХІДНОСТІ ПРОВЕДЕННЯ ДОСЛІДЖЕНЬ ЗА ОБРАНОЮ ТЕМОЮ	13
1.1 Аналіз існуючих моделей для оцінювання розміру програмного забезпечення ІС з кодом на РНР	13
1.2 Обґрунтування необхідності проведення досліджень за обраною темою	17
1.3 Висновки за розділом 1	18
2 НЕЛІНІЙНЕ РЕГРЕСІЙНУ МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ РОЗМІРУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ІС З КОДОМ НА РНР	20
2.1 Перевірка даних для побудови нелінійної регресійної моделі для оцінювання розміру програмного забезпечення ІС з кодом на РНР на наявність викидів	20
2.2 Побудова нелінійної регресійної моделі для оцінювання розміру програмного забезпечення ІС з кодом на РНР	24
2.3 Висновки за розділом 2	31
3 РОЗРОБКА ПРОЕКТУ ПРОГРАМИ ДЛЯ ОЦІНЮВАННЯ РОЗМІРУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ІС З КОДОМ НА РНР	33

3.1	Постановка задачі на розробку проекту програми для оцінювання розміру програмного забезпечення ІС з кодом на РНР	33
3.2	Розробка ескізного проекту програми для оцінювання розміру програмного забезпечення ІС з кодом на РНР	35
3.3	Розробка технічного проекту програми для оцінювання розміру програмного забезпечення ІС з кодом на РНР	42
3.4	Розробка робочого проекту програми для оцінювання розміру програмного забезпечення ІС з кодом на РНР	46
3.5	Висновки за розділом 3	56
4	РОЗРАХУНОК ЕКОНОМІЧНОЇ ЕФЕКТИВНОСТІ ВПРОВАДЖЕННЯ ПРОГРАМИ	58
4.1	Розрахунок витрат на створення й експлуатацію програми	58
4.2	Економічна ефективність розробки і впровадження програми	61
4.3	Висновки за розділом 4	63
5	ОХОРОНА ПРАЦІ	64
5.1	Аналіз існуючої проблеми охорони праці	64
5.2	Аналіз шкідливих та небезпечних факторів у офісному приміщенні з персональними комп'ютерами	65
5.3	Розрахунок системи пожежогасіння офісного приміщення	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 ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	85
ДОДАТОК А Технічне завдання на розробку програми для оцінювання розміру програмного забезпечення ІС з кодом на РНР	91
ДОДАТОК Б Текст програми	95
ДОДАТОК В Опис програми	104
ДОДАТОК Г Інструкція користувача	105
ДОДАТОК Д Програма та методика випробувань	107

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

ПЗ – програмне забезпечення;

COCOMO – the COnstructive COst MOdel;

HTML – hypertext markup language;

ISBSG – International Software Benchmarking Standards Group;

KLOC – the thousand lines of code;

LB – lower bound;

LOC – lines of code;

MD – the Mahalanobis distance;

MRE – a magnitude of relative error;

MMRE – a mean magnitude of relative error;

PHP – hypertext preprocessor;

PRED – percentage of prediction;

SQL – structured query language;

UB – upper bound.

ВСТУП

Актуальність теми. Оцінювання розміру програмних застосунків на початкових стадіях розробки є однією з важливих проблем менеджерів проектів при визначенні трудомісткості розробки програмного забезпечення (ПЗ) [1-3]. Невдале оцінювання розміру ПЗ часто є основною причиною хибних оцінок трудомісткості, тривалості та вартості і, як наслідок, невдалих програмних проектів.

Незважаючи на велику кількість існуючих на даний момент різноманітних методів і моделей для оцінювання розміру ПЗ [4-14], дослідження в цьому напрямку не припиняються [15-20]. Насамперед це пов'язано з низькою точністю оцінювання розмірів ПЗ на ранніх етапах його розробки. Одним із шляхів вирішення цієї проблеми є побудова відповідних моделей для оцінювання розміру ПЗ, яке розробляється як за допомогою певної мови програмування [4-17], так і для конкретного типу програмного застосунку [5, 6, 9-13, 16].

Хоча методи машинного навчання стають все більш популярними для оцінювання розміру ПЗ [19, 20], методи, засновані на нелінійному регресійному аналізі, ще не досягли повного потенціалу [21-23]. Ми застосовуємо метод побудови нелінійних регресійних моделей на основі багатовимірних нормалізуючих перетворень та інтервалів прогнозування [23]. У цьому методі інтервали прогнозування нелінійної регресії використовуються для виявлення викидів при побудові відповідної моделі. Зазвичай процес побудови нелінійної регресійної моделі є ітераційним, оскільки ми повторюємо побудову моделі для

нових даних після відкидання викидів. Якщо викидів немає, процес побудови нелінійної регресійної моделі завершується. Застосування цього методу, наприклад, в роботах [16, 17] дозволило отримати нелінійні регресійні моделі з добрими значеннями характеристик прогнозування таких як: множинний коефіцієнт детермінації, середня величина відносної похибки та відсоток прогнозованих значень. Тому ми будемо удосконалювати регресійну модель для оцінювання розміру ПЗ ІС з кодом на РНР порівняно з [13] саме із використанням зазначеного методу [23].

Мета і завдання дослідження. Метою роботи є підвищення достовірності оцінювання розміру ПЗ ІС з кодом на РНР за допомогою нелінійної регресійної моделі (НРМ).

Для досягнення поставленої мети потрібно вирішити такі завдання.

- Проаналізувати існуючі регресійні моделі для оцінювання розміру ПЗ ІС з кодом на РНР;
- Здійснити підбір сукупності даних, за якими буде будуватися НРМ для оцінювання розміру ПЗ ІС з кодом на РНР, та перевірити ці дані на наявність багатовимірних викидів;
- Удосконалити НРМ для оцінювання розміру ПЗ ІС з кодом на РНР на основі нормалізуючого перетворення;
- На підставі запропонованої НРМ розробити програму для оцінювання розміру ПЗ ІС з кодом на РНР.

Об'єктом дослідження є процес оцінювання розміру ПЗ ІС з кодом на РНР.

Предметом дослідження є регресійні моделі для оцінювання розміру ПЗ ІС з кодом на РНР.

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

Наукова новизна одержаних результатів. Удосконалено НРМ для оцінювання розміру ПЗ ІС з кодом на РНР на основі нормалізуючого перетворення Джонсона сімейства S_B із врахуванням викидів у регресії. Ця НРМ у порівнянні з існуючими регресійними рівняннями, як лінійними так і нелінійними, що побудовано без врахування 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 факторів.

Практичне значення одержаних результатів. Розроблено програму для оцінювання розміру ПЗ ІС з кодом на РНР. Програму створено scі-мовою для системи Scilab. Програму, що розроблено, можна використовувати для оцінювання розміру ПЗ ІС з кодом на РНР на ранніх стадіях проектування ПЗ відповідних застосунків.

Особистий внесок здобувача. У публікації [24], що зроблена у співавторстві, автору належить удосконалена НРМ для оцінювання розміру ПЗ ІС з кодом на РНР, яка була побудована під керівництвом керівника кваліфікаційної роботи.

Апробація результатів досліджень. Результати досліджень, викладені у роботі, були оприлюднені на одній конференції: II-й Всеукраїнській науково-практичній інтернет-конференції «Інформаційні технології: моделі, алгоритми, системи (ITMAS – 2021)» (Миколаїв, 26-28 жовтня 2021 р.).

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

1 АНАЛІЗ ІСНУЮЧИХ МОДЕЛЕЙ ДЛЯ ОЦІНЮВАННЯ РОЗМІРУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ІС З КОДОМ НА PHP ТА ОБҐРУНТУВАННЯ НЕОБХІДНОСТІ ПРОВЕДЕННЯ ДОСЛІДЖЕНЬ ЗА ОБРАНОЮ ТЕМОЮ

1.1 Аналіз існуючих моделей для оцінювання розміру програмного забезпечення ІС з кодом на PHP

У роботах [5, 6] Hee Beng Kuan Tan, Yuan Zhao та Hongyu Zhang запропонували лінійні регресійні рівняння для оцінювання розміру ПЗ (у тисячах строк коду, KLOC) ІС, що розробляються на базі певних мов програмування високого рівня, включаючи PHP. Запропоновані рівняння побудовано за допомогою множинного лінійного регресійного аналізу. У якості факторів (незалежних змінних) були використані метрики, що можуть бути отримані з діаграми класів: загальна кількість класів X_1 , загальна кількість зв'язків X_2 та середня кількість атрибутів на клас X_3 . Лінійне регресійне рівняння для оцінювання середнього розміру ПЗ ІС, що розробляються на основі PHP, з [5] має вигляд

$$\hat{Y} = \hat{b}_0 + \hat{b}_1 X_1 + \hat{b}_2 X_2 + \hat{b}_3 X_3, \quad (1.1)$$

де $\hat{Y}$ – оцінка середнього розміру ПЗ ІС з кодом на PHP у тисячах строк коду (KLOC); $\hat{b}_0$, $\hat{b}_1$, $\hat{b}_2$ та $\hat{b}_3$ – оцінки параметрів, які дорівнюють відповідно

-13,223, 1,241, 1,672 та 0,652.

Проте запропоноване в [5] лінійне регресійне рівняння (1.1) для оцінювання розміру ПЗ ІС з кодом на РНР має незадовільні значення показників MMRE і PRED(0,25), що дорівнюють 0,4919 та 0,5313 відповідно, для набору даних, за яким це рівняння було побудовано. Тому в роботі [13] було побудовано нелінійне регресійне рівняння із застосуванням чотиривимірного перетворення Джонсона сім'ї S_B на основі тих же трьох факторів (метрик діаграми класів), що і в [5]. Нелінійне регресійне рівняння для оцінювання середнього розміру ПЗ ІС, що розробляються на основі мови РНР, з [13] має вигляд

$$\hat{Y} = \hat{\phi}_Y + \hat{\lambda}_Y \left[1 + e^{-(\hat{Z}_Y - \hat{\gamma}_Y)/\hat{\eta}_Y} \right]^{-1}, \quad (1.2)$$

де $\hat{Z}_Y$ – оцінка середнього нормалізованого розміру за лінійним регресійним рівнянням

$$\hat{Z}_Y = \hat{b}_0 + \hat{b}_1 Z_1 + \hat{b}_2 Z_2 + \hat{b}_3 Z_3 \quad (1.3)$$

для даних, що нормалізовані за чотиривимірним нормалізуючим перетворенням Джонсона сімейства S_B , компоненти якого визначаються як

$$Z_j = \gamma_j + \eta_j \ln \frac{X_j - \phi_j}{\phi_j + \lambda_j - X_j}, \quad \phi_j < X_j < \phi_j + \lambda_j, \quad j=1,2,3. \quad (1.4)$$

В (1.4) Z_j – це стандартна гаусівська величина, $Z_j \sim N(0,1)$; γ_j , η_j , ϕ_j та λ_j – параметри нормалізуючого перетворення Джонсона сімейства S_B , $\eta_j > 0$, $\lambda_j > 0$. Залежна змінна Z_Y визначається аналогічно (1.4) з тією лише різницею, що замість Z_j , X_j , γ_j , η_j , ϕ_j та λ_j слід підставити відповідно Z_Y , Y , γ_Y , η_Y , ϕ_Y та λ_Y .

Оцінки параметрів чотиривимірного нормалізуючого перетворення Джонсона сімейства S_B є такими: $\hat{\gamma}_Y = 9,63091$, $\hat{\gamma}_1 = 15,5355$, $\hat{\gamma}_2 = 25,4294$, $\hat{\gamma}_3 = 0,72801$, $\hat{\eta}_Y = 1,05243$, $\hat{\eta}_1 = 1,58306$, $\hat{\eta}_2 = 2,54714$, $\hat{\eta}_3 = 0,54312$, $\hat{\phi}_Y = -1,4568$, $\hat{\phi}_1 = -1,8884$, $\hat{\phi}_2 = -6,9746$, $\hat{\phi}_3 = 3,2925$, $\hat{\lambda}_Y = 153102,605$, $\hat{\lambda}_1 = 243051,0$, $\hat{\lambda}_2 = 311229,5$ та $\hat{\lambda}_3 = 13,900$. Оцінки параметрів чотиривимірного нормалізуючого перетворення Джонсона сімейства S_B визначалися за методом максимальної правдоподібності. Оцінки параметрів рівняння (1.3) такі: $\hat{b}_0 = 1,02 \cdot 10^{-5}$, $\hat{b}_1 = 0,56085$, $\hat{b}_2 = 0,42491$, $\hat{b}_3 = 0,05846$. Оцінки параметрів рівняння (1.3) знаходилися за методом найменших квадратів.

Хоча запропоноване в [13] нелінійне регресійне рівняння (1.2) для оцінювання розміру ПЗ ІС з кодом на РНР має добре значення MMRE – 0,2199, але значення PRED(0,25) складає 0,7188, що є меншим за 0,75, яке вважається мінімальним прийнятним значенням. Крім того, як відомо [21, 23], регресійне рівняння на відміну від регресійної моделі не містить у своєму складі випадкової величини, яка описує похибки. А при оцінюванні розміру ПЗ як правило припускаються випадкових помилок. Тому виникає необхідність у подальшому удосконаленні нелінійного регресійного рівняння [13] для оцінювання розміру ПЗ ІС систем з кодом на РНР.

Побудову нелінійної регресійної моделі для оцінювання розміру ПЗ ІС з кодом на РНР ми будемо за методом [23], що заснований на багатовимірних нормалізуючих перетвореннях та інтервалах прогнозування. Суть цього методу [23] 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 [21]. На третьому етапі визначаються границі інтервалу прогнозування нелінійної регресії для рівня значущості, що дорівнює 0,05, за відповідним методом [21]. На останньому четвертому етапі перевіряють, чи є серед даних, за якими будувалася нелінійна регресійна модель такі, що виходять за границі інтервалу передбачення. 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. Як і в [13, 21, 23], у якості багатовимірного нормалізуючого перетворення ми використовуємо багатовимірне перетворення Джонсона для сімейства S_B , яке має вигляд:

$$\mathbf{Z} = \boldsymbol{\gamma} + \boldsymbol{\eta} \mathbf{h}[\boldsymbol{\lambda}^{-1}(\mathbf{X} - \boldsymbol{\phi})] \sim N_m(\mathbf{0}_m, \boldsymbol{\Sigma}), \quad (1.5)$$

where $\boldsymbol{\Sigma}$ – це коваріаційна матриця; $m = k + 1$; k – кількість факторів (у нашому випадку $k=3$); $\boldsymbol{\gamma}$, $\boldsymbol{\eta}$, $\boldsymbol{\phi}$ та $\boldsymbol{\lambda}$ – параметри перетворення (1.5); $\boldsymbol{\gamma} = (\gamma_1, \gamma_2, \dots, \gamma_m)^T$;

$\boldsymbol{\eta} = \text{diag}(\eta_1, \eta_2, \dots, \eta_m)$; $\boldsymbol{\lambda} = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_m)$; $\boldsymbol{\varphi} = (\varphi_1, \varphi_2, \dots, \varphi_m)^T$;
 $\mathbf{h}[(y_1, \dots, y_m)] = \{h_1(y_1), \dots, h_m(y_m)\}^T$; $h_i(\cdot)$ – одна з наступних функцій:

$$h = \begin{cases} \ln(y), & \text{для сімейства } S_L \text{ (log normal);} \\ \ln[y/(1-y)], & \text{для сімейства } S_B \text{ (bounded);} \\ \text{Arsh}(y), & \text{для сімейства } S_U \text{ (unbounded);} \\ y & \text{для сімейства } S_N \text{ (normal).} \end{cases} \quad (1.6)$$

Тут $y = (x - \varphi)/\lambda$; $\text{Arsh}(y) = \ln(y + \sqrt{y^2 + 1})$. Як і в [13], з (1.6) ми використовуємо функцію h для сімейства S_B .

1.2 Обґрунтування необхідності проведення досліджень за обраною темою

Як було зазначено в попередньому підрозділі, для оцінювання розміру ПЗ ІС з кодом на РНР існує лінійне регресійне рівняння [5, 6], яке побудовано за допомогою множинного лінійного регресійного аналізу на основі метрик діаграми класів. Проте запропоноване в [5] лінійне регресійне рівняння (1.1) для оцінювання розміру ПЗ ІС з кодом на РНР має незадовільні значення показників MMRE і PRED(0,25), що дорівнюють 0,4919 та 0,5313 відповідно, для набору даних, за яким це рівняння було побудовано. Тому в роботі [13] було побудовано нелінійне регресійне рівняння із застосуванням чотиривимірного перетворення Джонсона сімейства S_B на основі тих же трьох факторів (метрик діаграми класів), що і в [5].

Хоча запропоноване в [13] нелінійне регресійне рівняння (1.2) для оцінювання розміру ПЗ ІС з кодом на РНР має добре значення MMRE – 0,2199, але значення PRED(0,25) складає 0,7188, що є меншим за 0,75, яке вважається мінімальним прийнятним значенням. Крім того, як відомо [21, 23], регресійне рівняння на відміну від регресійної моделі не містить у своєму складі випадкової величини, яка описує похибки. А при оцінюванні розміру ПЗ як правило припускаються випадкових помилок. Тому виникає потреба у подальшому удосконаленні нелінійного регресійного рівняння [13] для оцінювання розміру ПЗ ІС систем з кодом на РНР.

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 розміру ПЗ ІС з кодом на РНР, здійснено обґрунтування необхідності проведення досліджень за обраною темою.

1) Зараз для оцінювання розміру ПЗ ІС з кодом на РНР використовуються методи як лінійного так нелінійного регресійного аналізу.

2) Для оцінювання розміру ПЗ ІС з кодом на РНР існують як лінійне так і нелінійне регресійні рівняння, які побудовано на основі метрик діаграми класів. Існуюче нелінійне регресійне рівняння для оцінювання розміру ПЗ ІС з кодом на РНР має значно кращі характеристики передбачення порівняно з лінійним рівнянням, але для нелінійного рівняння значення PRED(0,25) складає 0,7188, що є меншим за 0,75, яке вважається мінімальним прийнятним значенням. Крім

того регресійне рівняння на відміну від регресійної моделі не містить у своєму складі випадкової величини, яка описує похибки. А при оцінюванні розміру ПЗ як правило припускаються випадкових помилок. Тому виникає необхідність у подальшому удосконаленні нелінійного регресійного рівняння для оцінювання розміру ПЗ ІС систем з кодом на РНР.

3) Побудову нелінійної регресійної моделі для оцінювання розміру ПЗ ІС з кодом на РНР краще здійснювати за методом, що заснований на багатовимірних нормалізуючих перетвореннях та інтервалах прогнозування. Тому удосконалювати регресійну модель для оцінювання розміру ПЗ ІС з кодом на РНР доцільно саме на основі зазначеного методу.

2 НЕЛІНІЙНЕ РЕГРЕСІЙНЕ РІВНЯННЯ ДЛЯ ОЦІНЮВАННЯ РОЗМІРУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ІС З КОДОМ НА РНР

2.1 Перевірка даних для побудови нелінійної регресійної моделі для оцінювання розміру програмного забезпечення ІС з кодом на РНР на наявність викидів

Для побудови НРМ для оцінювання розміру ПЗ ІС з кодом на РНР використовується вибірка чотиривимірних негаусових даних з [5]: фактичний розмір ПЗ у тисячах рядках коду Y , загальна кількість класів X_1 , загальна кількість зв'язків X_2 та середня кількість атрибутів на клас X_3 в концептуальній моделі даних у вигляді діаграми класів з 32 ІС, розроблених з використанням мови програмування РНР. Ці дані наведені у таблиці 2.1.

Таблиця 2.1

Набір даних та квадрат відстані Махаланобіса (MD)

№	Y	X_1	X_2	X_3	квадрат MD	
					без перетворення	чотиривимірне перетворення
1	2	3	4	5	6	7
1	3,038	5	2	10,6	2,17	3,97
2	22,599	17	7	7	0,86	1,29
3	32,243	21	13	4,524	2,41	1,14

Продовження табл. 2.1

1	2	3	4	5	6	7
4	16,164	13	11	7,077	4,21	1,62
5	83,862	35	24	6,571	9,13	6,24
6	24,22	13	9	8,077	0,20	0,48
7	63,929	35	19	8,029	4,28	2,68
8	2,543	5	3	9,4	1,41	8,87
9	6,697	5	5	7	0,93	2,27
10	55,537	25	14	8,64	3,85	2,00
11	55,752	39	10	9,077	15,44	11,76
12	62,602	30	17	7	3,47	2,01
13	67,111	23	22	14,957	14,20	9,89
14	2,552	3	1	8,333	2,46	4,78
15	12,17	10	5	3,7	1,87	2,38
16	12,757	13	9	5	3,23	1,37
17	5,695	7	3	8,429	0,89	1,28
18	7,744	9	6	9,222	2,02	1,81
19	7,514	4	1	8	4,33	14,18
20	11,054	9	9	3,667	2,95	2,60
21	29,77	17	15	3,412	4,71	4,69
22	11,653	9	8	8,778	1,91	1,30
23	6,847	5	4	3,6	2,94	3,46
24	13,389	7	5	11,714	2,62	2,60
25	14,45	12	6	16,583	10,85	6,31
26	4,414	6	3	3,667	2,38	2,52
27	2,102	3	1	3,333	5,15	7,48
28	42,819	20	18	3,5	5,28	5,03
29	4,077	4	2	9	1,79	2,44
30	57,408	33	14	9,242	4,58	3,31
31	7,428	7	3	7	0,99	1,09
32	8,947	15	5	4	4,48	5,14

Для виявлення викидів у цих даних з таблиці 2.1 ми використовуємо метод, оснований на нормалізуючих перетвореннях і квадраті відстані Махаланобіса (Mahalanobis Distance) [25, 26], яку далі позначаємо як MD. Значення квадрату MD для кожної багатовимірної точки даних i , $i=1,2,\dots,N$, визначається як

$$d_i^2 = (\mathbf{Z}_i - \bar{\mathbf{Z}})^T S_N^{-1} (\mathbf{Z}_i - \bar{\mathbf{Z}}), \quad (2.1)$$

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

$$S_N = \frac{1}{N} \sum_{i=1}^N (\mathbf{Z}_i - \bar{\mathbf{Z}})(\mathbf{Z}_i - \bar{\mathbf{Z}})^T. \quad (2.2)$$

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 4.

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

$$\mathbf{Z} = \psi(\mathbf{X}). \quad (2.3)$$

В нашому випадку у якості перетворення (2.3) ми використовуємо багатовимірне перетворення Джонсона (1.5) для сімейства S_B .

Відомо [27], якщо дані мають багатовимірний нормальний розподіл, то розподіл квадрату MD поводить ся 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 (2.3), доки всі значення квадрату MD (2.1) будуть меншими чи дорівнювати квантилю розподілу χ^2 .

Для рівня значимості 0,005 та чотиривимірного перетворення Джонсона (1.5) для сімейства S_B викиди відсутні тому, що для всіх рядків даних квадрат MD є меншим ніж величина квантіля розподілу χ^2 , яка становить 14,86 для рівня значимості 0,005. В [5] також було зроблено припущення, що дані не містять викидів. Хоча застосування критерія на основі квадрата MD до даних з таблиці 2.1 без їх нормалізації показало, що дані системи 11 є чотиривимірним викидом, оскільки для цього рядка даних квадрат MD дорівнює 15,44 і є більшим ніж величина квантіля розподілу χ^2 , яка становить 14,86 для рівня значимості 0,005.

В подальшому чотиривимірні негаусові дані, що наведені у таблиці 2.1, використовуються для побудови нелінійної регресійної моделі для оцінювання розміру ПЗ ІС з кодом на РНР.

2.2 Побудова нелінійної регресійної моделі для оцінювання розміру програмного забезпечення ІС з кодом на РНР

Для побудови НРМ для оцінювання розміру ПЗ ІС з кодом на РНР спочатку чотиривимірні негаусівські дані з таблиці 2.1 ми нормалізуємо за допомогою чотиривимірного перетворення Джонсона для сімейства S_B . Далі для нормалізованих даних 32 рядків будемо лінійну регресійну модель

$$Z_Y = \hat{Z}_Y + \varepsilon = \hat{b}_0 + \hat{b}_1 Z_1 + \hat{b}_2 Z_2 + \hat{b}_3 Z_3 + \varepsilon, \quad (2.4)$$

де ε – це гаусівська випадкова величина, $\varepsilon \sim N(0, \sigma_\varepsilon^2)$; $\hat{\sigma}_\varepsilon$ – це оцінка середнього квадратичного відхилення, значення якої дорівнює 0,28865.

Параметри рівняння (2.4) оцінювалися за методом найменших квадратів. Оцінки параметрів $\hat{b}_0$, $\hat{b}_1$, $\hat{b}_2$, та $\hat{b}_3$ відповідно дорівнюють 0, 0,523808, 0,452016 та 0,065708.

Далі будемо НРМ за лінійним рівнянням (2.4) та чотиривимірним перетворенням Джонсона для сімейства S_B

$$Y = \hat{\phi}_Y + \hat{\lambda}_Y \left[1 + e^{-(\hat{Z}_Y + \varepsilon - \hat{\gamma}_Y) / \hat{\eta}_Y} \right]^{-1}, \quad (2.5)$$

де $\hat{Z}_Y = \hat{b}_0 + \hat{b}_1 Z_1 + \hat{b}_2 Z_2 + \hat{b}_3 Z_3$; змінні Z_1 , Z_2 та Z_3 визначаються за (1.4).

Оцінки параметрів чотиривимірного нормалізуючого перетворення Джонсона сімейства S_B γ_Y , $\hat{\gamma}_1$, $\hat{\gamma}_2$, $\hat{\gamma}_3$, η_Y , $\hat{\eta}_1$, $\hat{\eta}_2$, $\hat{\eta}_3$, ϕ_Y , $\hat{\phi}_1$, $\hat{\phi}_2$, $\hat{\phi}_3$, λ_Y , $\hat{\lambda}_1$, $\hat{\lambda}_2$ та $\hat{\lambda}_3$ відповідно дорівнюють 1,05371, 1,09823, 0,93190, 0,760326, 0,57530, 0,71545,

0,81509, 0,59083, 1,602, 2,32299, -0,07664, 3,233, 95,0607, 48,7648, 29,5328, 14,1365. Оцінки параметрів чотиривимірного нормалізуючого перетворення Джонсона сімейства S_B визначалися за методом максимальної правдоподібності.

Далі згідно з [23], на третьому етапі ми визначаємо границі інтервалу передбачення нелінійної регресії для рівня значущості, що дорівнює 0,05. Згідно з [21], границі інтервалу передбачення нелінійної регресії визначаються як

$$\psi_Y^{-1} \left(\hat{Z}_Y \pm t_{\alpha/2, v} S_{Z_Y} \left\{ 1 + \frac{1}{N} + (\mathbf{z}_X^+)^T \left[(\mathbf{Z}_X^+)^T \mathbf{Z}_X^+ \right]^{-1} (\mathbf{z}_X^+) \right\}^{1/2} \right), \quad (2.6)$$

де ψ_Y^{-1} – зворотне до нормалізуючого перетворення ψ_Y для залежної змінної Y ; $\hat{Z}_Y$ – результат передбачення 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 $k=3$); $\mathbf{Z}_X^+$ – матриця центрованих нормалізованих незалежних змінних, яка містить значення $Z_{1i} - \bar{Z}_1, Z_{2i} - \bar{Z}_2, \dots, Z_{ki} - \bar{Z}_k$; $t_{\alpha/2, v}$ – квантиль t -розподілу Стюдента в залежності від рівня значимості $\alpha/2$ та кількості ступенів свободи v ;

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

$$(\mathbf{Z}_X^+)^T \mathbf{Z}_X^+ = \begin{pmatrix} S_{Z_1 Z_1} & S_{Z_1 Z_2} & \dots & S_{Z_1 Z_k} \\ S_{Z_2 Z_1} & S_{Z_2 Z_2} & \dots & S_{Z_2 Z_k} \\ \dots & \dots & \dots & \dots \\ S_{Z_k Z_1} & S_{Z_k Z_2} & \dots & S_{Z_k Z_k} \end{pmatrix}, \quad (2.7)$$

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

В (2.7) знаки мінус та плюс відповідають нижній та верхній границям інтервалу передбачення нелінійної регресії, що побудована на основі нормалізуючого перетворення.

Матриця $(\mathbf{Z}_X^+)^T \mathbf{Z}_X^+$ для нормалізованих за багатовимірним перетворенням Джонсона сімейства S_B 32 рядків даних з таблиці 2.1 є такою

$$(\mathbf{Z}_X^+)^T \mathbf{Z}_X^+ = \begin{pmatrix} 32,00 & 28,54 & 4,65 \\ 28,54 & 32,00 & 2,10 \\ 4,65 & 2,10 & 32,00 \end{pmatrix}.$$

Границі інтервалу передбачення нелінійної регресії, що отримані за допомогою чотиривимірного перетворення Джонсона сімейства S_B на двох ітераціях, наведені у табл.2.2.

Для НРМ (2.5) з оцінками параметрів, що отримані для 32 рядків даних з таблиці 2.1, виявилось: значення Y для двох ІС (рядки 8 і 19) виходять за визначені межі інтервалу передбачення. В таблиці 2.2 ліва границя інтервалу передбачення на першій ітерації позначена як LB_1 , а права – як UB_1 .

Згідно з [23], ми відкинули рядки 8 і 19 як чотиривимірні викиди та перейшли до другої ітерації побудови нелінійної регресійної моделі.

Таблиця 2.2 – Границі інтервалу передбачення нелінійної регресії, що отримані за допомогою чотиривимірного перетворення Джонсона сімейства S_B на двох ітераціях

№	Y	LB ₁	UB ₁	LB ₂	UB ₂
1	2	3	4	5	6
1	3,038	2,504	9,651	2,866	6,682
2	22,599	8,893	43,205	11,406	32,205
3	32,243	15,776	60,564	20,821	50,797
4	16,164	9,922	46,171	13,451	36,722
5	83,862	52,484	89,449	66,389	91,410
6	24,22	8,763	41,979	11,641	32,389
7	63,929	40,864	84,610	51,053	82,590
8	2,543	2,773	11,615	-	-
9	6,697	3,188	15,010	4,077	11,164
10	55,537	21,760	69,583	28,256	61,414
11	55,752	26,915	79,943	33,610	72,454
12	62,602	30,516	78,218	39,047	73,095
13	67,111	34,000	83,624	47,109	82,140
14	2,552	1,816	3,771	1,926	2,998
15	12,17	4,463	23,480	5,612	16,259
16	12,757	8,205	39,966	10,863	30,485
17	5,695	3,304	15,408	3,966	10,552
18	7,744	5,313	27,317	6,898	19,909
19	7,514	1,996	5,423	-	-
20	11,054	5,691	30,501	7,676	22,812
21	29,77	12,913	57,249	17,868	47,590
22	11,653	6,188	32,050	8,309	24,183
23	6,847	2,726	11,585	3,321	8,417
24	13,389	4,185	21,494	5,386	15,532

Продовження таблиці 2.2

1	2	3	4	5	6
25	14,45	7,101	39,716	9,381	29,042
26	4,414	2,764	11,783	3,265	8,141
27	2,102	1,754	3,305	1,835	2,665
28	42,819	18,477	67,669	25,645	60,008
29	4,077	2,240	7,425	2,550	5,445
30	57,408	29,078	77,897	36,555	71,581
31	7,428	3,255	15,010	3,897	10,272
32	8,947	5,890	32,405	7,397	22,281

На другій ітерації, згідно з [23], ми знову нормалізували дані з 30 рядків з таблиці 2.1 (за винятком відкинутих рядків 8 і 19) за допомогою чотиривимірного нормалізуючого перетворення Джонсона сімейства S_B . Для даних з 30 рядків з таблиці 2.1 оцінки параметрів чотиривимірного нормалізуючого перетворення Джонсона сімейства S_B γ_Y , $\hat{\gamma}_1$, $\hat{\gamma}_2$, $\hat{\gamma}_3$, η_Y , $\hat{\eta}_1$, $\hat{\eta}_2$, $\hat{\eta}_3$, $\hat{\phi}_Y$, $\hat{\phi}_1$, $\hat{\phi}_2$, $\hat{\phi}_3$, λ_Y , $\hat{\lambda}_1$, $\hat{\lambda}_2$ та $\hat{\lambda}_3$ відповідно дорівнюють 1,09175, 1,00719, 0,81228, 0,78962, 0,60232, 0,74733, 0,77582, 0,58146, 1,602, 2,08914, 0,24715, 3,233, 101,3795, 48,0019, 28,0864, 14,2491. Оцінки параметрів чотиривимірного нормалізуючого перетворення Джонсона сімейства S_B визначалися за методом максимальної правдоподібності.

Для виявлення викидів у даних з 30 рядків з таблиці 2.1 (за винятком відкинутих рядків 8 і 19) ми знову використовуємо метод, оснований на нормалізуючих перетвореннях і квадраті MD. Для рівня значимості 0,005 та чотиривимірного перетворення Джонсона (1.5) для сімейства S_B викиди відсутні тому, що для всіх зазначених 30 рядків даних квадрат MD є меншим ніж величина квантіля розподілу χ^2 , яка становить 14,86 для рівня значимості 0,005.

Далі для нормалізованих даних з 30 рядків з таблиці 2.1 (за винятком відкинутих рядків 8 і 19) будуємо лінійну регресійну модель (2.4) та нелінійну регресійну модель (2.5), оцінки параметрів для яких є такими. Оцінки параметрів $\hat{b}_0$, $\hat{b}_1$, $\hat{b}_2$, та $\hat{b}_3$ відповідно дорівнюють 0, 0,472755, 0,526228 та 0,072036, а оцінка середньо квадратичного відхилення має значення 0,18735.

Далі згідно з [23], на третьому етапі другої ітерації за формулою (2.7) ми визначаємо границі інтервалу передбачення нелінійної регресії для рівня значущості, що дорівнює 0,05. Значення лівої та правої границі інтервалу передбачення нелінійної регресії на другій ітерації наведені в таблиці 2.2. В таблиці 2.2 ліва границя інтервалу передбачення на другій ітерації позначена як LB₂, а права – як UB₂. Для НРМ (2.5) з оцінками параметрів, що отримані для 30 рядків даних з таблиці 2.1 (за винятком відкинутих рядків 8 і 19), виявилось: значення Y для всіх зазначених рядків даних не виходять за визначені межі інтервалу передбачення. Тому, згідно з [23], ітерації завершуються, шукана НРМ (2.5) з оцінками параметрів, що отримані для 30 рядків даних з таблиці 2.1 (за винятком відкинутих рядків 8 і 19) вважається побудованою.

Для визначення точності оцінювання розміру ПЗ ІС з кодом на РНР за допомогою побудованої НРМ (2.5) з оцінками параметрів, що отримані для 30 рядків даних з таблиці 2.1 (за винятком відкинутих рядків 8 і 19), були використані відомі показники точності прогнозування, такі як множинний або вибірковий коефіцієнт детермінації R^2 , середня величина відносної похибки MMRE та відсоток прогнозування на рівні величини відносної помилки 0,25, PRED(0,25). Ці показники зазвичай застосовують для оцінювання якості прогнозування за допомогою регресійних моделей і в інженерії програмного

забезпечення [28, 29]. Допустимі значення MMRE і PRED(0,25) складають не більше 0,25 і не менше 0,75 відповідно.

Вибірковий коефіцієнт детермінації R^2 визначається 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), $SS_T = \sum_{i=1}^N (Y_i - \bar{Y})^2$. Нагадаємо, 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 .

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

$$MMRE = \frac{1}{N} \sum_{i=1}^N MRE_i.$$

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

Значення $PRED(0,25)$ обчислюється як

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

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

Для побудованої НРМ (2.5) для оцінювання розміру ПЗ ІС з кодом на РНР значення R^2 , $MMRE$ та $PRED(0,25)$ дорівнюють відповідно 0,9885, 0,1725 та 0,7667. Нелінійна регресійна модель (2.5) порівняно з нелінійним регресійним рівнянням (1.2) має на 27 % менше значення $MMRE$ та більші значення R^2 та $PRED(0,25)$ відповідно на 0,2 % та 6,2 %. Значення R^2 , $MMRE$ та $PRED(0,25)$ для моделі (2.5) вказують на добрі результати оцінювання розміру ПЗ ІС з кодом на РНР за її допомогою.

2.3 Висновки за розділом 2

У даному розділі виконано перевірку даних для побудови нелінійної регресійної моделі для оцінювання розміру ПЗ ІС з кодом на РНР на наявність викидів, побудовано нелінійну регресійну модель для оцінювання розміру ПЗ ІС з кодом на РНР.

1) Перевірку даних для побудови нелінійної регресійної моделі для оцінювання розміру ПЗ ІС з кодом на РНР на наявність викидів виконано за методом, який оснований на нормалізуючих перетвореннях і квадраті відстані

Махаланобіса (Mahalanobis Distance). З'ясовано, що за методом, який оснований на нормалізуючих перетвореннях і квадраті відстані Махаланобіса, чотиривимірні дані не містять багатовимірних викидів. І навпаки, застосування критерія на основі квадрата відстані Махаланобіса до даних без їх нормалізації показало, що дані містять багатомірний викид.

2) Удосконалено нелінійну регресійну модель для оцінювання розміру ПЗ ІС з кодом на РНР на основі нормалізуючого перетворення Джонсона сімейства S_B із врахуванням викидів у регресії. Ця нелінійна регресійна модель у порівнянні з існуючими регресійними рівняннями, як лінійними так і нелінійними, що побудовано 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 факторів.

3) Отримані результати свідчать про те, що для побудови нелінійних регресійних моделей для оцінювання розміру ПЗ ІС з кодом на РНР, потрібно використовувати нормалізуючі перетворення з урахуванням викидів у нелінійній регресії.

3 РОЗРОБКА ПРОЕКТУ ПРОГРАМИ ДЛЯ ОЦІНЮВАННЯ РОЗМІРУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ІС З КОДОМ НА РНР

3.1 Постановка задачі на розробку проекту програми для оцінювання розміру програмного забезпечення ІС з кодом на РНР

Виходячи з аналізу існуючих моделей для оцінювання розміру ПЗ ІС з кодом на РНР, сформулюємо постановку задачі. Потрібно створити програму для оцінювання розміру ПЗ ІС з кодом на РНР, 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 ІС з кодом на РНР: перша метрика Y включає фактичний розмір ПЗ в тисячах строк коду (KLOC), друга X_1 та третя X_2 метрики визначають відповідно загальну кількість класів і загальну кількість співвідношень у концептуальній моделі даних у вигляді діаграми класів, а останньою четвертою метрикою X_3 є середня кількість атрибутів на клас, тобто відношення загальної кількості атрибутів у концептуальній моделі даних до X_1 .

2) Вводити з інтерфейсу програми довірчу ймовірність, три метрики концептуальної моделі даних у вигляді діаграми класів, за якими потрібно

оцінити розмір ПЗ ІС з кодом на РНР в тисячах строк коду: загальну кількість класів X_1 і загальну кількість співвідношень X_2 у концептуальній моделі даних у вигляді діаграми класів, середню кількість атрибутів на клас X_3 , тобто відношення загальної кількості атрибутів у концептуальній моделі даних до X_1 .

3) Виконувати нормалізацію емпіричних даних за нормалізуючим перетворенням Джонсона сімейства S_B .

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

5) Здійснювати побудову НРМ.

6) Обчислювати значення коефіцієнту детермінації R^2 .

7) Здійснювати побудову інтервалу передбачення нелінійної регресії.

8) Виводити на екран нелінійну регресію, довірчий інтервал нелінійної регресії розміру ПЗ ІС з кодом на РНР в тисячах строк коду, значення коефіцієнту детермінації R^2 .

9) Виводити у текстовий файл результати оцінювання розміру ПЗ ІС з кодом на РНР в тисячах строк коду.

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 розміру програмного забезпечення ІС з кодом на РНР

В ескізному проекті було виявлено варіанти використання та побудовано їх сценарії, розроблено зовнішню специфікацію програми для оцінювання розміру ПЗ ІС з кодом на РНР 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.

Діаграма варіантів використання програми для оцінювання розміру ПЗ ІС з кодом на РНР зображена на рисунку 3.1.

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

Таблиця 3.1 Виявлені варіанти використання

Основні актори	Найменування	Формулювання
Користувач	Ввести дані метрик ПЗ ІС з кодом на РНР	Вводить дані про фактичний розмір ПЗ ІС з кодом на РНР в тисячах строк коду, загальну кількість класів, загальну кількість співвідношень і середню кількість атрибутів на клас у концептуальній моделі даних.
Користувач	Ввести три метрики концептуальної моделі даних	Вводить три метрики концептуальної моделі даних, за якими потрібно оцінити розмір ПЗ ІС з кодом на РНР з кодом в тисячах строк коду: загальну кількість класів, загальну кількість співвідношень і середню кількість атрибутів на клас у концептуальній моделі даних.
Користувач	Ввести довірчу ймовірність	Вводить довірчу ймовірність, для якої потрібно оцінити довірчий інтервал та інтервал прогнозування нелінійної регресії розміру ПЗ ІС з кодом на РНР в тисячах строк коду.
Користувач	Нормалізувати дані	Виконує нормалізацію внесених даних
Користувач	Побудувати нелінійну регресійну модель	Будує нелінійну регресійну модель для оцінювання значення розміру ПЗ ІС з кодом на РНР
Користувач	Отримати довірчий інтервал та інтервал прогнозування	Отримує довірчий інтервал та інтервал прогнозування нелінійної регресії для інтервального оцінювання вибіркового середнього розміру ПЗ ІС з кодом на РНР із заданою ймовірністю

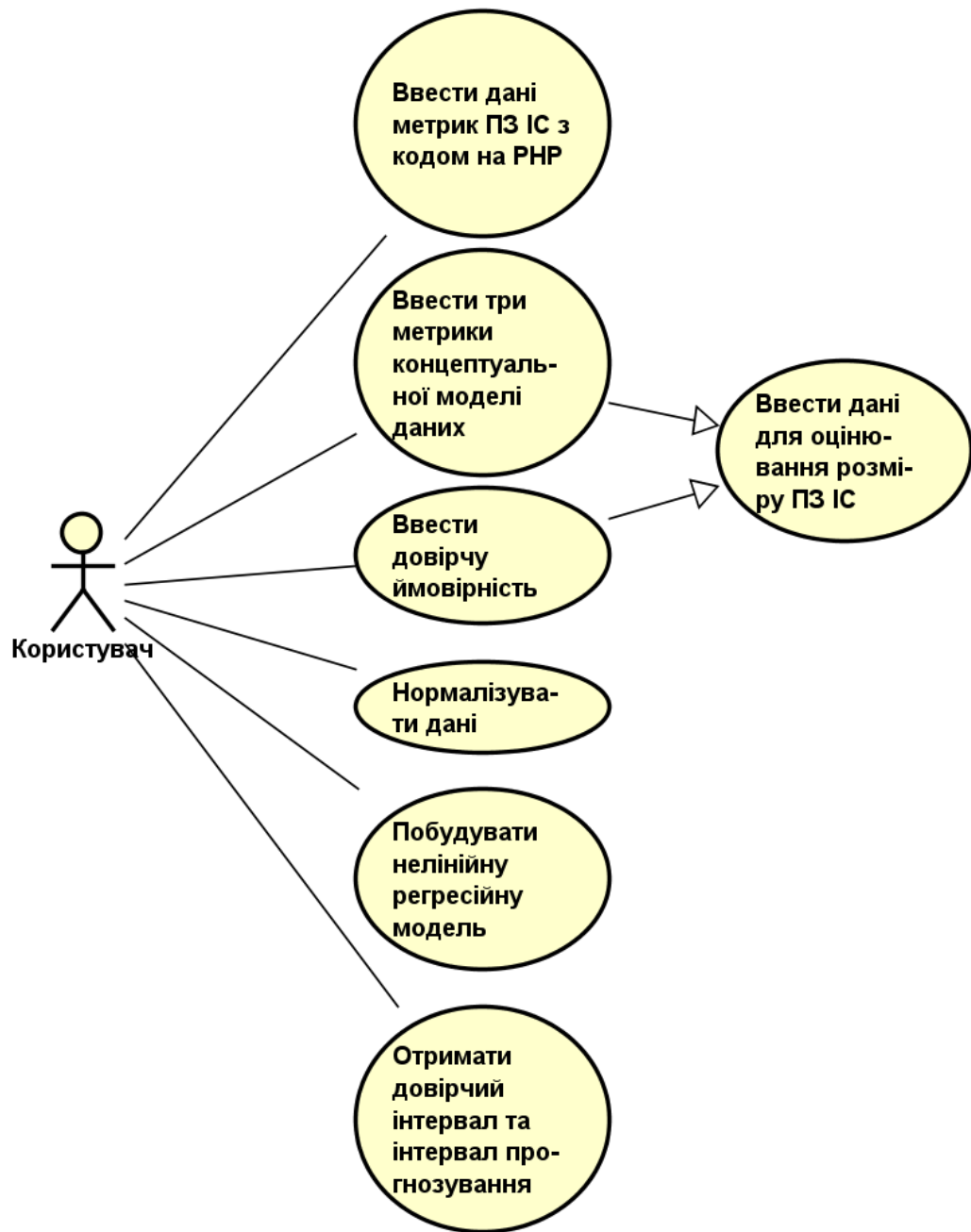


Рисунок 3.1 – Діаграма варіантів використання програми для оцінювання розміру ПЗ ІС з кодом на РНР



Рисунок 3.2 – Діаграма діяльності ПЗ для оцінювання розміру ПЗ ІС з кодом на PHP

Зовнішня специфікація програми.

Назва програми: PHPSoftSizeEst.

Функції, які виконує програма:

1) Вводити з текстового файлу емпіричні дані з метрик ПЗ ІС з кодом на РНР: перша метрика Y включає фактичний розмір ПЗ в тисячах строк коду (KLOC), друга X_1 та третя X_2 метрики визначають відповідно загальну кількість класів і загальну кількість співвідношень у концептуальній моделі даних, а останньою четвертою метрикою X_3 є середня кількість атрибутів на клас, тобто відношення загальної кількості атрибутів у концептуальній моделі даних до X_1 .

2) Вводити з інтерфейсу програми довірчу ймовірність, три метрики концептуальної моделі даних, за якими потрібно оцінити розмір ПЗ ІС з кодом на РНР в тисячах строк коду: загальну кількість класів X_1 і загальну кількість співвідношень X_2 у концептуальній моделі даних, середню кількість атрибутів на клас X_3 , тобто відношення загальної кількості атрибутів у концептуальній моделі даних до X_1 .

3) Виконувати нормалізацію емпіричних даних за нормалізуючим перетворенням Джонсона сімейства S_B .

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

5) Здійснювати побудову НРМ.

6) Обчислювати значення коефіцієнту детермінації R^2 .

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

8) Виводити на екран нелінійну регресію, довірчий інтервал та інтервал прогнозування нелінійної регресії розміру ПЗ ІС з кодом на РНР в тисячах строк коду, значення коефіцієнту детермінації R^2 .

9) Виводити у текстовий файл результати оцінювання розміру ПЗ ІС з кодом на РНР у тисячах строк коду.

Вхідні дані:

- емпіричні дані з метрик ПЗ ІС з кодом на РНР: перша метрика Y включає фактичний розмір ПЗ в тисячах строк коду (KLOC), друга X_1 та третя X_2 метрики визначають відповідно загальну кількість класів і загальну кількість співвідношень у концептуальній моделі даних, а останньою четвертою метрикою X_3 є середня кількість атрибутів на клас, тобто відношення загальної кількості атрибутів у концептуальній моделі даних до X_1 ;

- значення довірчої ймовірності, трьох метрик концептуальної моделі даних, за якими потрібно оцінити розмір ПЗ ІС з кодом на РНР (в тисячах строк коду): загальну кількість класів X_1 і загальну кількість співвідношень X_2 у концептуальній моделі даних, середню кількість атрибутів на клас X_3 , тобто відношення загальної кількості атрибутів у концептуальній моделі даних до X_1 .

Вихідні дані:

- значення нелінійної регресії, границь довірчого інтервалу та інтервалу прогнозування нелінійної регресії розміру ПЗ ІС з кодом на РНР.

- значення коефіцієнту детермінації.

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 розміру ПЗ ІС з кодом на РНР, вихід з програми (закриття всіх графічних вікон).

Для проектування інтерфейсу програми для оцінювання розміру ПЗ ІС з кодом на РНР представимо стани програми при виконанні у вигляді відповідної діаграми станів. Діаграма станів програми для оцінювання розміру ПЗ ІС з кодом на РНР наведена на рисунку 3.3.

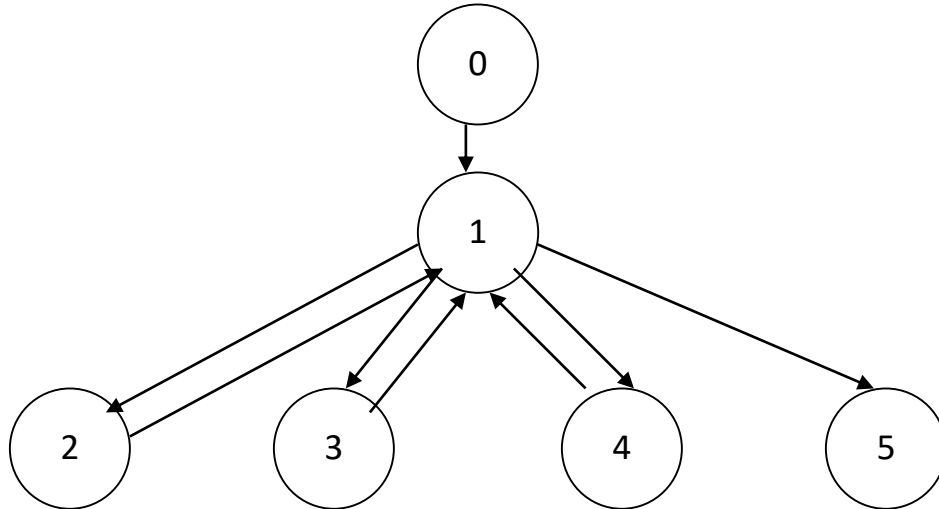


Рисунок 3.3 – Діаграма станів програми для оцінювання розміру ПЗ ІС з кодом на РНР

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 інтервалу прогнозування нелінійної регресії розміру ПЗ ІС з кодом на РНР, 5 – кінцевий стан.

3.3 Розробка технічного проекту програми для оцінювання розміру програмного забезпечення ІС з кодом на PHP

В технічному проекті було спроектовано програму для оцінювання розміру ПЗ ІС з кодом на PHP на основі покрокової деталізації зверху вниз, розроблено специфікації модулів програми, побудовано алгоритми модулів програми.

Програма складається з таких модулів: головного модуля (PHPSoftSizeEst), модуля для введення емпіричних даних з файлу та їх візуалізації (DataInput), модуля для побудови нелінійної регресії і довірчого інтервалу та інтервалу прогнозування нелінійної регресії розміру ПЗ ІС з кодом на PHP (PHPSoftSizeConfIntEst), модуля для закриття графічних вікон (W_Close). Схема взаємодії модулів програми наведена на рисунку 3.4.

Специфікація головного модуля.

Назва модуля: PHPSoftSizeEst.

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 розміру ПЗ та введення його значення у тисячах строк коду.
- 5) Виклик функції для визначення нелінійної регресії, довірчого інтервалу та інтервалу прогнозування нелінійної регресії розміру ПЗ ІС з кодом на PHP.
- 6) Виклик функції для закриття графічних вікон.

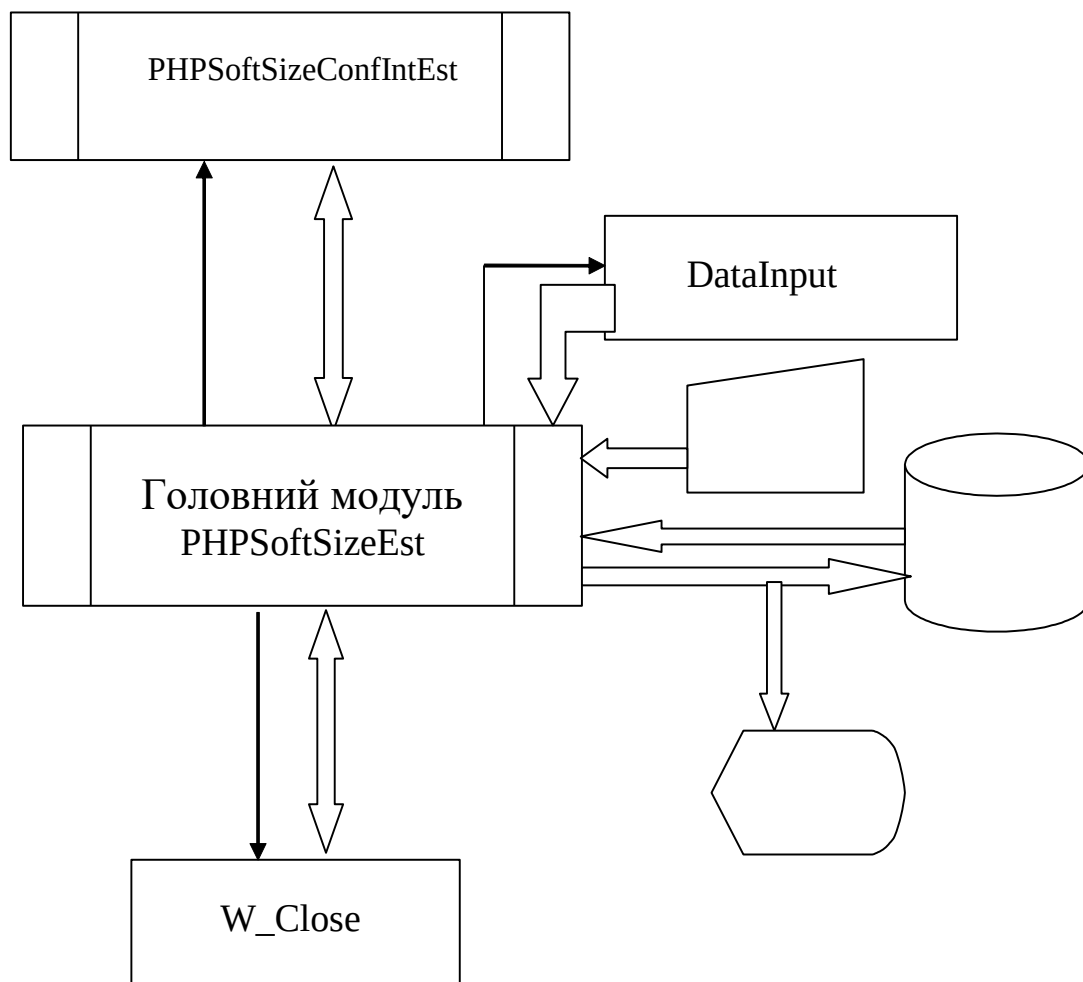


Рисунок 3.4 – Схема взаємодії модулів програми

Вхідні дані:

- емпіричні дані з метрик ПЗ ІС з кодом на РНР: перша метрика Y включає фактичний розмір ПЗ в тисячах строк коду (KLOC), друга X_1 та третя X_2 метрики визначають відповідно загальну кількість класів і загальну кількість співвідношень у концептуальній моделі даних, а останньою четвертою метрикою X_3 є середня кількість атрибутів на клас, тобто відношення загальної кількості атрибутів у концептуальній моделі даних до X_1 ;

- значення довірчої ймовірності, трьох метрик концептуальної моделі даних, за якими потрібно оцінити розмір ПЗ ІС з кодом на РНР (в тисячах строк коду): загальну кількість класів X_1 і загальну кількість співвідношень X_2 у концептуальній моделі даних, середню кількість атрибутів на клас X_3 , тобто відношення загальної кількості атрибутів у концептуальній моделі даних до X_1 .

Вихідні дані:

- значення нелінійної регресії та границь довірчого інтервалу та інтервалу прогнозування нелінійної регресії розміру ПЗ ІС з кодом на РНР.

- значення коефіцієнту детермінації.

Зовнішні ефекти: поява графічного вікна для вводу даних, управління виводом даних, побудови довірчого інтервалу та інтервалу прогнозування нелінійної регресії розміру ПЗ ІС з кодом на РНР, 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 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);

- ідентифікатор графічного вікна.

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

Специфікація модуля для визначення нелінійної регресії та довірчого інтервалу нелінійної регресії розміру ПЗ ІС з кодом на PHP.

Назва модуля: PHPSoftSizeConfIntEst.

Функції, які виконує модуль:

1) Визначає нелінійну регресію та довірчий інтервал нелінійної регресії розміру ПЗ ІС з кодом на PHP.

2) Обчислює значення коефіцієнту детермінації для нелінійної регресії.

Вхідні дані:

- двомірні масиви значень випадкових величини (емпіричних даних);

- кількість значень випадкових величин (емпіричних даних).

Вихідні дані:

- значення нелінійної регресії, нижньої та верхньої границь довірчого інтервалу нелінійної регресії розміру ПЗ ІС з кодом на РНР;

- значення коефіцієнту детермінації для нелінійної регресії.

Зовнішні ефекти: з'являються результати нелінійної регресії, нижньої та верхньої границь довірчого інтервалу нелінійної регресії розміру ПЗ ІС з кодом на РНР та виводиться значення коефіцієнту детермінації для нелінійної регресії.

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 розміру програмного забезпечення ІС з кодом на РНР

В робочому проекті було розроблено програму для оцінювання розміру ПЗ ІС з кодом на РНР, здійснено тестування програми, розроблено програмну документацію, здійснено випробування програми.

Спочатку sci-мовою системи моделювання Scilab було виконано кодування алгоритмів, що були розроблені на стадії технічного проекту. Текст програми наведений в додатку Б. Приклад кодування модуля

PHPSoftSizeConfIntEst, що здійснює оцінювання розміру ПЗ ІС з кодом на PHP, sci-мовою в системі моделювання Scilab наведено нижче.

```
//Функція оцінювання розміру ПЗ інформаційних систем з кодом на PHP
function PHPSoftSizeEst(x, yx, n)
//Зчитуємо значення змінних із текстових полів та
//перетворюємо їх до числового типу.
x1i=eval(get(edit_text4,'string'));
x2i=eval(get(edit_text5,'string'));
x3i=eval(get(edit_text6,'string'));
Pconf=eval(get(edit_text7,'string'));

x1A=0; x2A=0; x3A=0; yA=0;
for i=1:n
    x1A=x1A+x(i,1); x2A=x2A+x(i,2); x3A=x3A+x(i,3); yA=yA+yx(i,1);
end;
x1A=x1A/n; x2A=x2A/n; x3A=x3A/n; yA=yA/n;
Sx1x1=0; Sx2x2=0; Sx3x3=0;
Sx1x2=0; Sx1x3=0; Sx2x3=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);
    Sx1x3=Sx1x3+(x(i,1)-x1A)*(x(i,3)-x3A);
    Sx2x3=Sx2x3+(x(i,2)-x2A)*(x(i,3)-x3A);
    Sx2x2=Sx2x2+(x(i,2)-x2A)*(x(i,2)-x2A);
    Sx3x3=Sx3x3+(x(i,3)-x3A)*(x(i,3)-x3A);
    SyA=SyA+(yx(i,1)-yA)*(yx(i,1)-yA);
end;
Sxx(1,1)=Sx1x1; Sxx(1,2)=Sx1x2; Sxx(1,3)=Sx1x3;
Sxx(2,1)=Sx1x2; Sxx(2,2)=Sx2x2; Sxx(2,3)=Sx2x3;
Sxx(3,1)=Sx1x3; Sxx(3,2)=Sx2x3; Sxx(3,3)=Sx3x3;

Sxx_1=1/Sxx;

ay=1.091752366476;
ax1=1.007190; ax2=0.812284514764; ax3=0.7896157118;
by=0.60232130;
bx1=0.74733212763; bx2=0.77582060; bx3=0.581457617658;
cy=1.602;
cx1=2.08914350; cx2=0.247154038; cx3=3.233;
dy=101.3795327;
dx1=48.0019355; dx2=28.086461764; dx3=14.249170;

z1A=0; z2A=0; z3A=0;
for i=1:n
```

```

    z1A=z1A+ax1+bx1*log((x(i,1)-cx1)/(dx1-x(i,1)+cx1));
    z2A=z2A+ax2+bx2*log((x(i,2)-cx2)/(dx2-x(i,2)+cx2));
    z3A=z3A+ax3+bx3*log((x(i,3)-cx3)/(dx3-x(i,3)+cx3));
    //z1A=z1A+log10(x(i,1)); z2A=z2A+log10(x(i,2));
    z3A=z3A+log10(x(i,3));
end;
z1A=z1A/n; z2A=z2A/n; z3A=z3A/n;
Sz1z1=0; Sz2z2=0; Sz3z3=0;
Sz1z2=0; Sz1z3=0; Sz2z3=0;
for i=1:n
    Sz1z1=Sz1z1+(ax1+bx1*log((x(i,1)-cx1)/(dx1-x(i,1)+cx1))-
    z1A)*(ax1+bx1*log((x(i,1)-cx1)/(dx1-x(i,1)+cx1))-z1A);
    Sz1z2=Sz1z2+(ax1+bx1*log((x(i,1)-cx1)/(dx1-x(i,1)+cx1))-
    z1A)*(ax2+bx2*log((x(i,2)-cx2)/(dx2-x(i,2)+cx2))-z2A);
    Sz1z3=Sz1z3+(ax1+bx1*log((x(i,1)-cx1)/(dx1-x(i,1)+cx1))-
    z1A)*(ax3+bx3*log((x(i,3)-cx3)/(dx3-x(i,3)+cx3))-z3A);
    Sz2z3=Sz2z3+(ax2+bx2*log((x(i,2)-cx2)/(dx2-x(i,2)+cx2))-
    z2A)*(ax3+bx3*log((x(i,3)-cx3)/(dx3-x(i,3)+cx3))-z3A);
    Sz2z2=Sz2z2+(ax2+bx2*log((x(i,2)-cx2)/(dx2-x(i,2)+cx2))-
    z2A)*(ax2+bx2*log((x(i,2)-cx2)/(dx2-x(i,2)+cx2))-z2A);
    Sz3z3=Sz3z3+(ax3+bx3*log((x(i,3)-cx3)/(dx3-x(i,3)+cx3))-
    z3A)*(ax3+bx3*log((x(i,3)-cx3)/(dx3-x(i,3)+cx3))-z3A);
    //Sz1z1=Sz1z1+(log10(x(i,1))-z1A)*(log10(x(i,1))-z1A);
    //Sz1z2=Sz1z2+(log10(x(i,1))-z1A)*(log10(x(i,2))-z2A);
    //Sz1z3=Sz1z3+(log10(x(i,1))-z1A)*(log10(x(i,3))-z3A);
    //Sz2z3=Sz2z3+(log10(x(i,2))-z2A)*(log10(x(i,3))-z3A);
    //Sz2z2=Sz2z2+(log10(x(i,2))-z2A)*(log10(x(i,2))-z2A);
    //Sz3z3=Sz3z3+(log10(x(i,3))-z3A)*(log10(x(i,3))-z3A);
end;
Szz(1,1)=Sz1z1; Szz(1,2)=Sz1z2; Szz(1,3)=Sz1z3;
Szz(2,1)=Sz1z2; Szz(2,2)=Sz2z2; Szz(2,3)=Sz2z3;
Szz(3,1)=Sz1z3; Szz(3,2)=Sz2z3; Szz(3,3)=Sz3z3;

Szz_1=1/Szz;

zi(1)=ax1+bx1*log((x1i-cx1)/(dx1-x1i+cx1))-z1A;
zi(2)=ax2+bx2*log((x2i-cx2)/(dx2-x2i+cx2))-z2A;
zi(3)=ax3+bx3*log((x3i-cx3)/(dx3-x3i+cx3))-z3A;
//zi(1)=log10(x1i)-z1A; zi(2)=log10(x2i)-z2A; zi(3)=log10(x3i)-z3A;
ziT=zi';
ziSzz_1=ziT*Szz_1;
result=ziSzz_1*zi;

Df=n-4;
alpha=1-Pconf;

```



```

P=1-alpha/2;
//P=0.975;
Q=1-P;

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

b0=0;
b1=0.47275475;
b2=0.5262282833;
b3=0.07203592772;

Szyzyn4=0; Syy=0;
for i=1:n
    z_y=ay+by*log((yx(i,1)-cy)/(dy-yx(i,1)+cy));
    z_x1=ax1+bx1*log((x(i,1)-cx1)/(dx1-x(i,1)+cx1));
    z_x2=ax2+bx2*log((x(i,2)-cx2)/(dx2-x(i,2)+cx2));
    z_x3=ax3+bx3*log((x(i,3)-cx3)/(dx3-x(i,3)+cx3));
    Szyzyn4=Szyzyn4+(b0+b1*z_x1+b2*z_x2+b3*z_x3-z_y)^2;

    //Szyzyn4=Szyzyn4+..
    //(b0+b1*log10(x(i,1))+b2*log10(x(i,2))+b3*log10(x(i,3))-
    log10(yx(i,1)))^2;

    zy_result=b0+b1*z_x1+b2*z_x2+b3*z_x3;
    Syy=Syy+(yx(i,1)-(cy+dy/(1+exp(-(zy_result-ay)/by))))^2;
    //Syy=Syy+(yx(i,1)-10^b0*x(i,1)^b1*x(i,2)^b2*x(i,3)^b3)^2;
end;
Szyzyn4=Szyzyn4/(n-4)
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;

z_x1=ax1+bx1*log((x(1,1)-cx1)/(dx1-x(1,1)+cx1));
z_x2=ax2+bx2*log((x(1,2)-cx2)/(dx2-x(1,2)+cx2));
z_x3=ax3+bx3*log((x(1,3)-cx3)/(dx3-x(1,3)+cx3));
zy_result=b0+b1*z_x1+b2*z_x2+b3*z_x3;

//zy_result=b0+b1*log10(x1i)+b2*log10(x2i)+b3*log10(x3i);

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

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

y_result=cy+dy/(1+exp(-(zy_result-ay)/by));

```

```

yLBC=cy+dy/(1+exp(-(zyLBC-ay)/by));
yUBC=cy+dy/(1+exp(-(zyUBC-ay)/by));
yLBPr=cy+dy/(1+exp(-(zyLBPr-ay)/by));
yUBPr=cy+dy/(1+exp(-(zyUBPr-ay)/by));

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

file_PC5=mopen('D:\Data_F\PHP_IS_30_data5.txt','w+');
//for i=1:3
    mfprintf(file_PC5, "%5.6f ", yUBC);
//end
fclose(file_PC5);

file_PC4=mopen('D:\Data_F\PHP_IS_30_data4.txt','w+');
for i = 1:3
    for j = 1:3 mfprintf(file_PC4, "%5.6f ", Szz_1(i,j)); end;
    mfprintf(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 радио_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 sse-файл розміром 17 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.3.5.

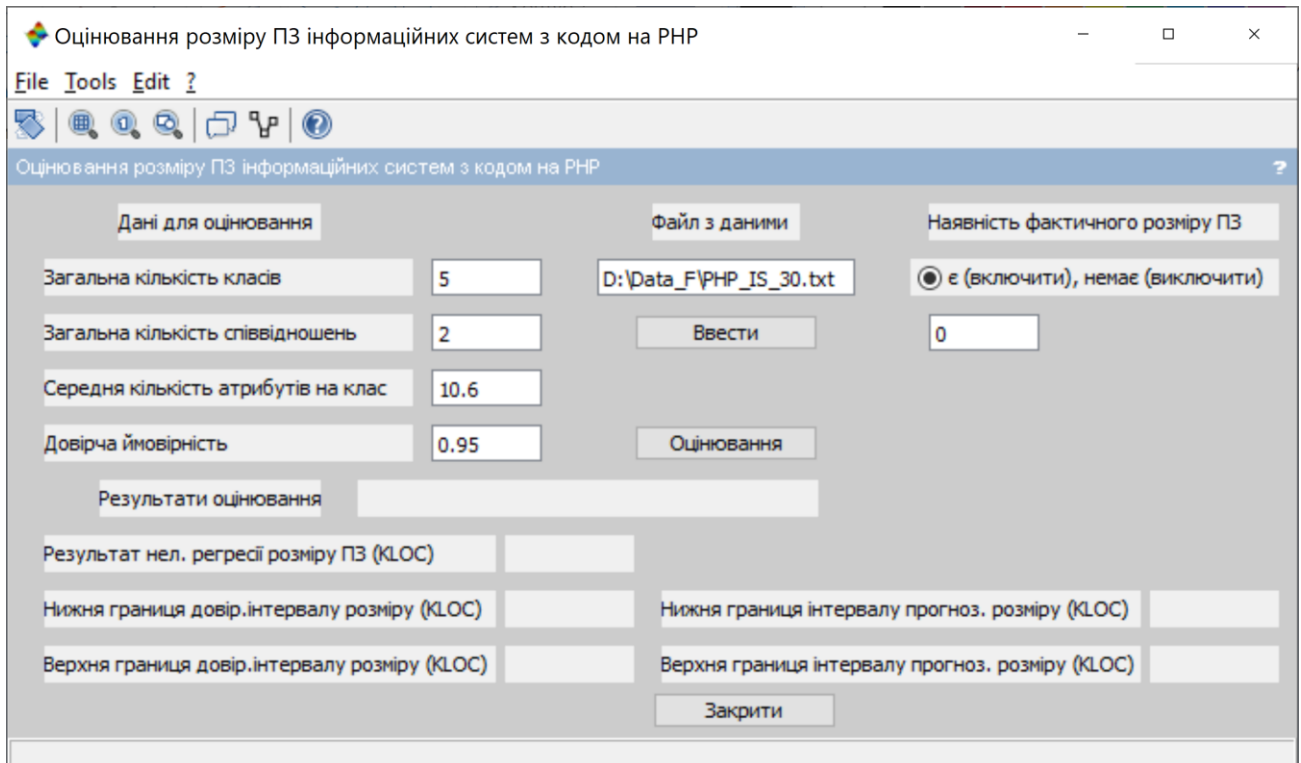


Рисунок 3.5. – Графічне вікно програми із інтерфейсними компонентами

Було проведено тестування окремих модулів. У якості прикладу наведемо результати тестування для модуля оцінювання розміру ПЗ ІС з кодом на PHP

для емпіричних даних – кількості строк коду (у тисячах строк коду) 30 ІС в залежності від загальної кількості класів, загальної кількості співвідношень у концептуальній моделі даних та середньої кількості атрибутів на клас, тобто відношення загальної кількості атрибутів у концептуальній моделі даних до загальної кількості класів. Було побудовано НРМ на основі нормалізуючого перетворення Джонсона сімейства S_B . Результати візуалізації емпіричних даних про кількість строк коду (у тисячах строк коду) 30 ІС в залежності від загальної кількості класів наведені на рис.3.6.

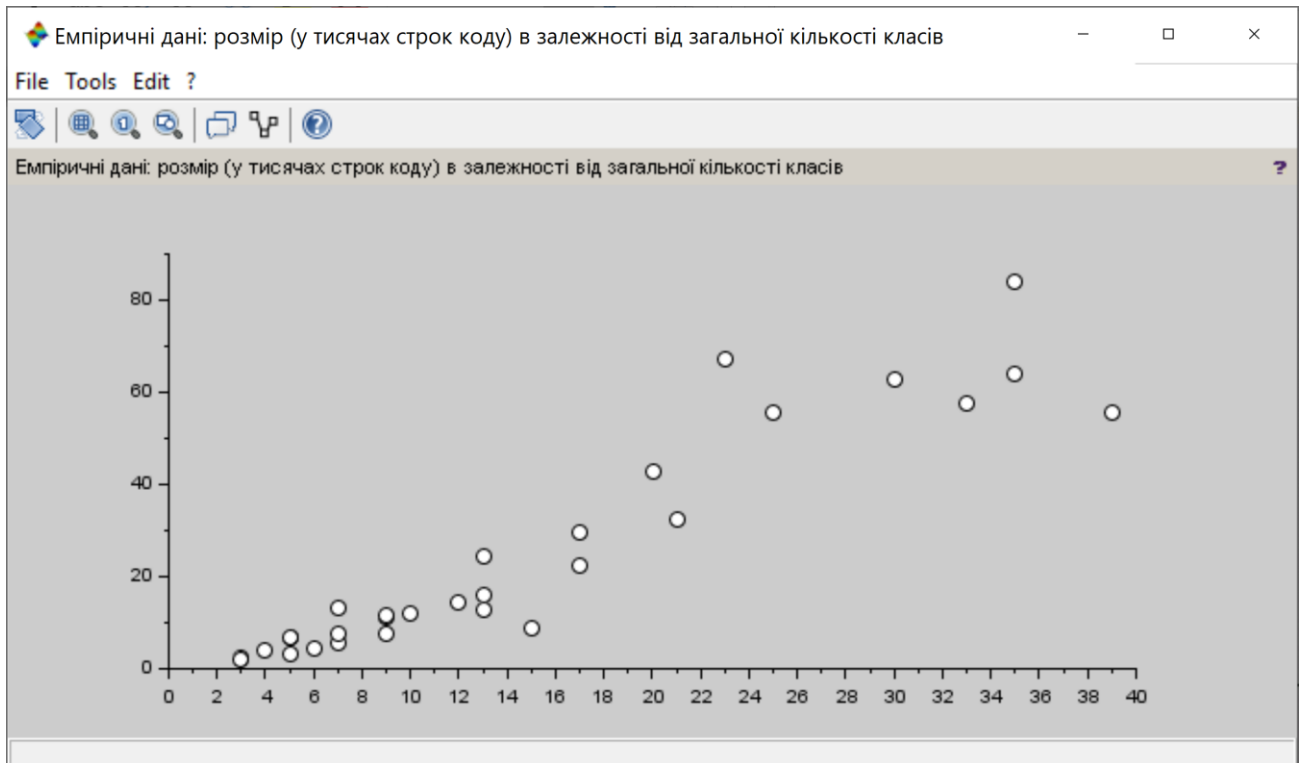


Рисунок 3.6. – Емпіричні дані про кількість строк коду (у тисячах строк коду) 30 ІС в залежності від загальної кількості класів

Результати візуалізації емпіричних даних про кількість строк коду (у тисячах строк коду) 30 ІС в залежності від загальної кількості співвідношень у концептуальній моделі даних наведені на рис.3.7.

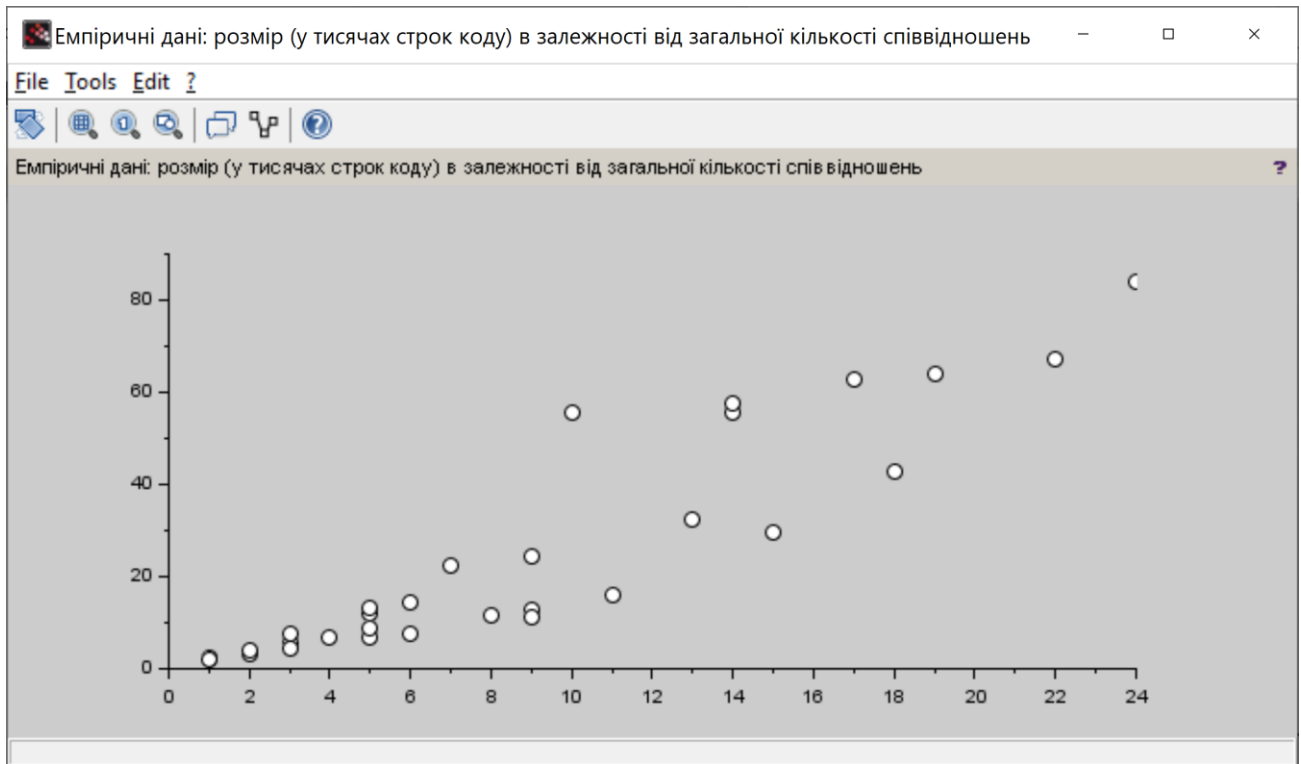


Рисунок 3.7. – Емпіричні дані про кількість строк коду (у тисячах строк коду) 30 ІС в залежності від загальної кількості співвідношень у концептуальній моделі даних

Результати візуалізації емпіричних даних про кількість строк коду (у тисячах строк коду) 30 ІС в залежності від середньої кількості атрибутів на клас, тобто відношення загальної кількості атрибутів у концептуальній моделі даних до загальної кількості класів, наведені на рис.3.8.

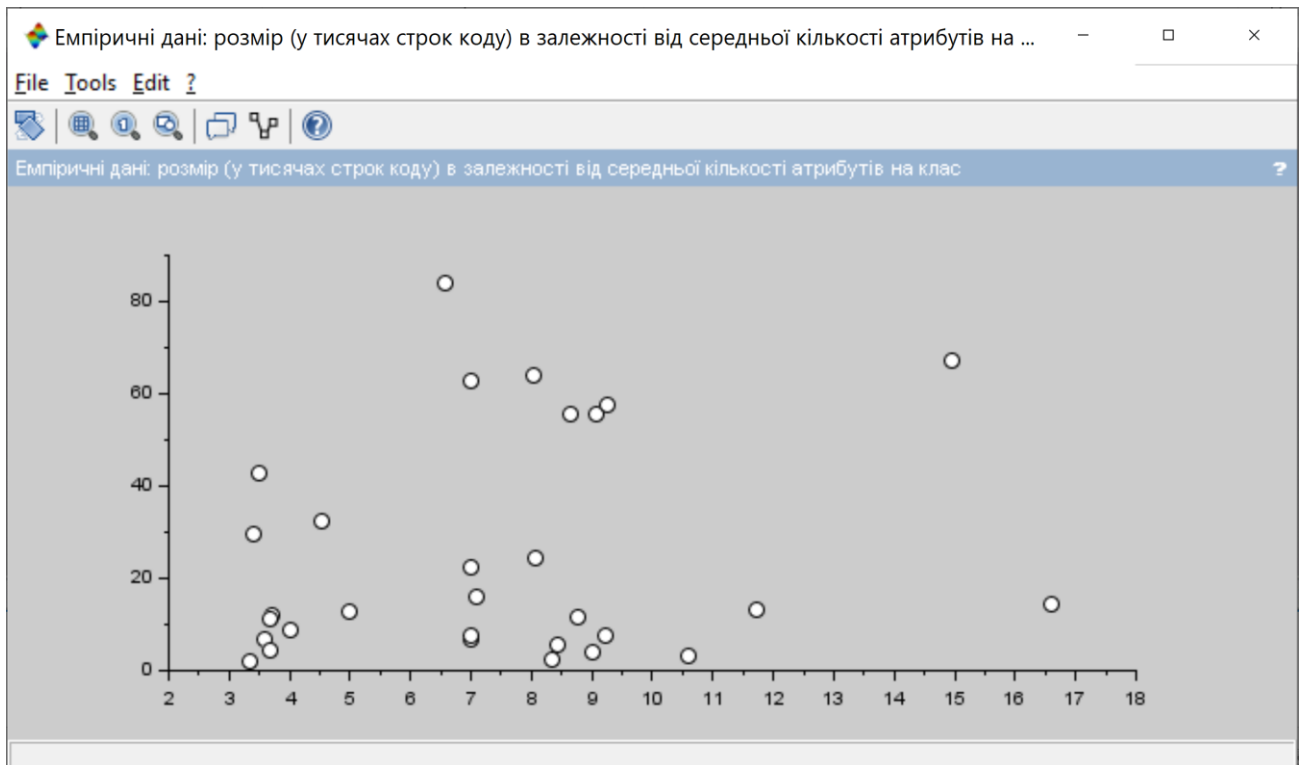


Рисунок 3.8. – Емпіричні дані про кількість строк коду (у тисячах строк коду)
30 ІС в залежності від середньої кількості атрибутів на клас
у концептуальній моделі даних

Результати оцінювання значень нелінійної регресії, нижньої і верхньої границь довірчого інтервалу нелінійної регресії розміру ПЗ ІС з кодом на РНР та результати обчислення коефіцієнту детермінації R^2 наведені на рис.3.9. Також у разі відомого фактичного розміру ПЗ ІС може бути обчислено значення відносної похибки (MRE) результату оцінювання значень нелінійної регресії в порівнянні з фактичним розміром ПЗ ІС з кодом на РНР для зазначених метрик.

Оцінювання розміру ПЗ інформаційних систем з кодом на PHP

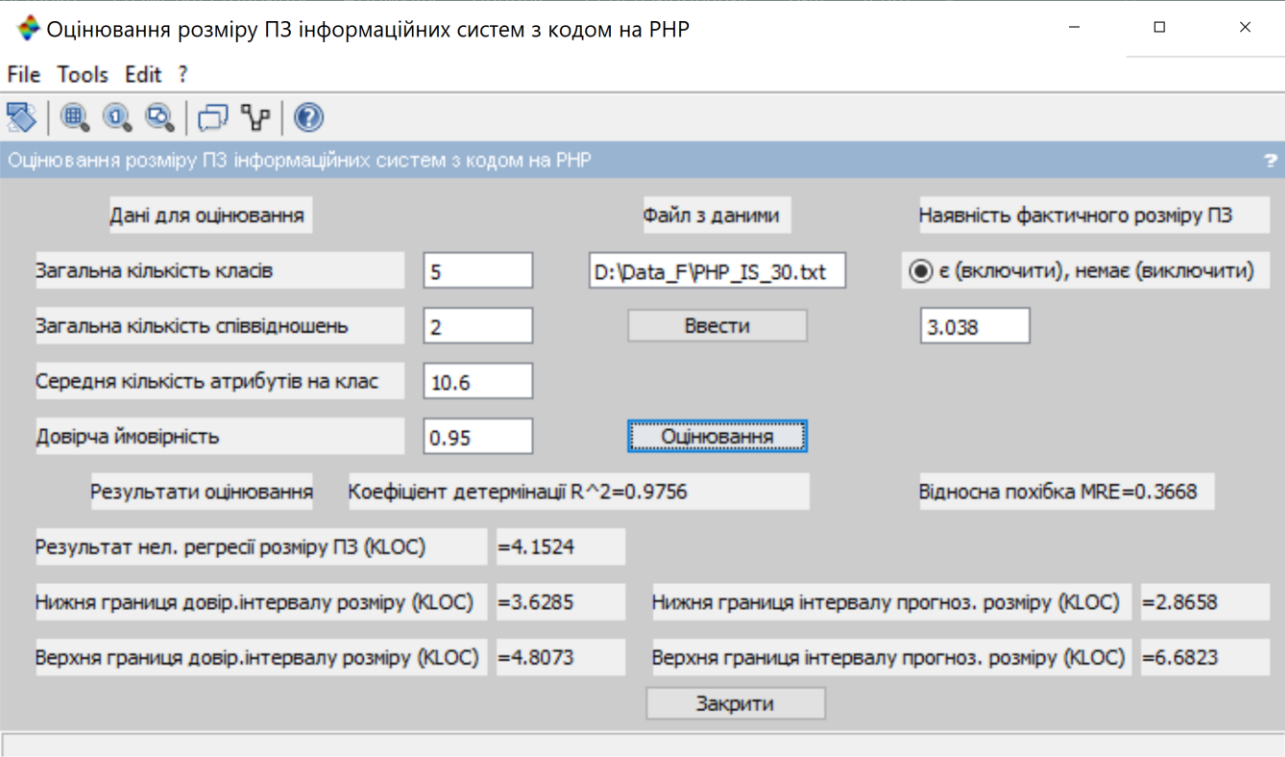
File Tools Edit ?

Оцінювання розміру ПЗ інформаційних систем з кодом на PHP

Дані для оцінювання		Файл з даними	Наявність фактичного розміру ПЗ	
Загальна кількість класів	5	D:\Data_F\PHP_IS_30.txt	☐ є (включити), немає (виключити)	
Загальна кількість співвідношень	2	Ввести	0	
Середня кількість атрибутів на клас	10.6			
Довірча ймовірність	0.95	Оцінювання		
Результати оцінювання		Коефіцієнт детермінації $R^2=0.9756$		
Результат нел. регресії розміру ПЗ (KLOC)	=4.1524			
Нижня границя довір.інтервалу розміру (KLOC)	=3.6285	Нижня границя інтервалу прогноз. розміру (KLOC)	=2.8658	
Верхня границя довір.інтервалу розміру (KLOC)	=4.8073	Верхня границя інтервалу прогноз. розміру (KLOC)	=6.6823	
Закрити				

Рисунок 3.9. – Результати оцінювання значень нелінійної регресії, нижньої і верхньої границь довірчого інтервалу нелінійної регресії розміру ПЗ ІС з кодом на PHP

На рис.3.10 наведено значення відносної похибки (MRE) результату оцінювання значень нелінійної регресії в порівнянні з фактичним розміром ПЗ ІС з кодом на PHP для зазначених метрик.



Оцінювання розміру ПЗ інформаційних систем з кодом на PHP

File Tools Edit ?

Оцінювання розміру ПЗ інформаційних систем з кодом на PHP

Дані для оцінювання		Файл з даними	Наявність фактичного розміру ПЗ
Загальна кількість класів	5	D:\Data_F\PHP_IS_30.txt	☒ є (включити), немає (виключити)
Загальна кількість співвідношень	2	Ввести	3.038
Середня кількість атрибутів на клас	10.6		
Довірча ймовірність	0.95	Оцінювання	
Результати оцінювання		Коефіцієнт детермінації $R^2=0.9756$	
Відносна похибка MRE=0.3668			
Результат нел. регресії розміру ПЗ (KLOC)		=4.1524	
Нижня границя довір.інтервалу розміру (KLOC)		=3.6285	
Верхня границя довір.інтервалу розміру (KLOC)		=4.8073	
Нижня границя інтервалу прогноз. розміру (KLOC)		=2.8658	
Верхня границя інтервалу прогноз. розміру (KLOC)		=6.6823	
Закрити			

Рисунок 3.10. – Результати визначення відносної похибки оцінювання розміру ПЗ ІС з кодом на PHP

Отримані результати практично співпали з результатами, що наведені у табл.2.2 та оцінками інших авторів. Це свідчить про працездатність розробленої програми.

3.5 Висновки за розділом 3

У даному розділі виконано постановку задачі на розробку проекту програми для оцінювання розміру ПЗ ІС з кодом на PHP, розроблено ескізний,

технічний та робочий проекту програми для оцінювання розміру ПЗ ІС з кодом на РНР.

1) В ескізному проекті було виявлено варіанти використання та побудовано їх сценарії, розроблено зовнішню специфікацію програми для оцінювання розміру ПЗ ІС з кодом на РНР та стани цієї програми при виконанні.

2) В технічному проекті було спроектовано програму для оцінювання розміру ПЗ ІС з кодом на РНР на основі покрокової деталізації зверху вниз, розроблено специфікації модулів програми, побудовано алгоритми модулів програми.

3) В робочому проекті було розроблено програму для оцінювання розміру ПЗ ІС з кодом на РНР, здійснено тестування програми, розроблено програмну документацію, здійснено випробування програми. Результати випробування програми практично співпали 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 15000 грн. Додаткова заробітна плата складає 20% від основної. Виходячи з цього, основна і додаткова заробітна плата програміста

3000 грн/мес, а вартість сучасного ПК складає 16800 грн. (приведена вартість комп'ютера на базі процесора Intel Celeron Dual-Core J1800 (2.41 ГГц) / RAM 4 ГБ / HDD 500 ГБ / Intel HD Graphics / LAN / Windows 8 SL та монітора 18.5" Asus VS197DE). 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	90,00
Заправлення картриджа до принтера	90,00
CD-диск	6,00
Непередбачені витрати	150,00
Разом матеріали і комплектуючі	336,00

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

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

Таблиця 4.2 – Витрати на розробку ПЗ

Найменування витрат	Одиниця	Кількість
Тривалість розробки	Мес	1,5
Основна і додаткова ЗП	Грн.	18000
Відрахування на соціальні заходи	Грн.	6616,8
Загальногосподарські витрати	Грн.	250
Витрати на матеріали	Грн.	336,00
Витрати на електроенергію	Грн.	151,06

Відповідно до формули (4.1) вартість програми:

$$\text{Спр} = (18000 + 6616,8 + 250 + 151,06) * 1,5 + 336,00 = 25353,86 \text{ грн.}$$

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

$$\text{Аоб} = 16800 * 0,6 = 10080 \text{ грн.}$$

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

$$\text{Вм} = 16800 * 0,05 = 840 \text{ грн.}$$

Річний обсяг робіт ПК у годинах визначається в такий спосіб:

$$\Phi_{\text{м}} = 264,5 * T_{\text{з}}, \quad (4.2)$$

де $T_{\text{з}}$ – це середнє місячне завантаження устаткування (близько 4 годин); 264,5 – середня кількість робочих днів у році.

Отже, річний обсяг роботи ПК складе:

$$\Phi_{\text{м}} = 264,5 * 4 = 1058 \text{ годин.}$$

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

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

Експлуатаційні витрати для ПК за рік складуть:

$$Z_{зр} = 10080 + 840 + 908,08 = 11828,08 \text{ грн.}$$

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

$$Z_{се} = 25353,86 + 11828,08 = 37181,94 \text{ грн.}$$

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:

$$(15000 \cdot 12) \cdot 0,6 = 108000 \text{ грн.}$$

Річний економічний ефект розраховується по формулі:

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

де ΔC_n – вивільнені кошти після впровадження програми (108000 грн.) мінус експлуатаційні витрати (11828,08 грн.); E_n – коефіцієнт ефективності (дорівнює коефіцієнту амортизації – 0,6); k – одноразові витрати на впровадження програми (25353,86 грн.).

Річний економічний ефект буде

$$E_p = 96171,92 - 25353,86 \cdot 0,6 = 80959,60 \text{ грн.}$$

Строк окупності програми розраховується по формулі:

$$T = \frac{k}{\Delta C_n} = \frac{25353,86}{96171,92} = 0,26 \text{ (року)}.$$

Отже строк окупності програми складає приблизно 3,2 місяця.

4.3 Висновки за розділом 4

У даному розділі виконано розрахунок витрат на створення й експлуатацію програми для оцінювання розміру програмного забезпечення інформаційних систем з кодом на РНР та економічної ефективності її розробки і впровадження.

4) Витрати на створення й експлуатацію програми для оцінювання розміру програмного забезпечення інформаційних систем з кодом на РНР складають майже 37,2 тис. грн.

5) Економічна ефективність розробки і впровадження програми оцінюється за такими показниками як річний економічний ефект та строк окупності програми, які склали майже 81 тис. грн та 3,2 місяці відповідно.

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»). 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 у офісному приміщенні з персональними комп'ютерами

Небезпечним називається виробничий фактор, вплив якого на працюючу людину в певних умовах приведе до травми або іншому раптовому різкому погіршенню здоров'я. Якщо ж виробничий фактор приведе до захворювання або зниження працездатності, то його вважають шкідливим. Залежно від рівня й тривалості впливу шкідливий виробничий фактор може стати небезпечним.

На офісного працівника при роботі впливають дві основні групи шкідливих факторів:

1) Фізичні, до яких відносять фактори, що породжуються безпосереднім впливом оточуючого середовища (електромагнітне випромінювання, освітленість робочого місця, температура повітря, шум).

2) Психофізіологічні, до яких відносять фізичне та нервово-психічне перевантаження організму (зорове напруження, статичні навантаження, нервово-психічне навантаження).

Освітлення – отримання, розподіл та використання світлової енергії для забезпечення нормальних умов праці. Світло, крім забезпечення фізичної можливості розрізняти предмети праці, також впливає на психічний стан людини та перебіг фізіологічних процесів в організмі. Раціонально влаштоване освітлення виробничих приміщень позитивно впливає на працівників, підвищує ефективність та безпеку праці, знижує втому та травматизм, забезпечує високу працездатність.

За вимогами санітарних норм освітлення повинно бути достатньо яскравим, рівномірним, не засліплювати очі та не створювати відблисків на робочій поверхні. За спектральним складом освітлення має наближатися до сонячного світла. Гігієнічними нормами вимагається максимально використовувати природне освітлення, оскільки денне світло краще сприймається органами зору. Штучне освітлення у виробничих та адміністративно-громадських приміщеннях має здійснюватись системою загального рівномірного освітлення, допускають застосування системи комбінованого освітлення. Значення освітленості на поверхні робочого столу в зоні розміщення документів має становити 300-500лк. Як джерела світла для штучного освітлення мають застосовувати переважно люмінесцентні лампи типу ЛБ.

Одним з найнесприятливіших факторів, що зменшує працездатність та уважність робітників, створює передумови до травматизму та захворювань, є шум. Інтенсивний шум та вібрації негативно впливають не лише на слух, а і на

нервову, серцево-судинну, та більшість інших систем організму, викликаючи зниження слуху, гіпертонію, психічні розлади тощо.

Гігієнічними нормативами (СН 3222-85, ГОСТ 12.1.003-85, ГР 2411-81) встановлені допустимі рівні звуку різних частот для робітників певних професій. У таблиці 5.1 наведені нормативи для людей, що працюють з ЕОМ.

Таблиця 5.1 – Допустимі рівні шуму

Вид трудової діяльності	Рівні звукового тиску в дБ октавних смугах із середньо геометричними частотами, ГЦ									
	31,5	63	125	250	500	1000	2000	4000	8000	Рівні звуку, еквівалентні рівні звуку, дБа/дБАекв
Програмісти ЕОМ	86	71	61	54	49	45	42	40	38	50
Оператори в залах обробки інформації на ЕОМ ті оператори комп'ютерного набору	96	83	74	68	63	60	57	55	54	65
В приміщеннях для розташування шумних агрегатів ЕОМ	103	91	83	77	73	70	68	66	64	75

Виходячи з даних норм, при проведенні комплексу заходів зі зменшення шуму усувається не будь-яка віброакустична активність обладнання, а тільки та, яка перевищує встановлений граничний рівень та шкідливо впливає на організм людини.

Усунення промислового шуму в першу чергу відбувається шляхом вдосконалення технологічних процесів та обладнання. При цьому в першу чергу усуваються найбільш сильні джерела шуму. Також необхідно знешумлювати усе обладнання, яке цього потребує, оскільки знешумлення окремих одиниць при наявності великої кількості обладнання, що продовжує випромінювати шум, недоцільно.

Самопочуття та працездатність людини також сильно залежать від теплового стану організму, на який впливають такі фізичні фактори виробничого середовища, як температура повітря, вологість, швидкість руху повітря, атмосферний тиск, доля кисню у повітрі тощо. Наприклад, якщо кількість кисню в повітрі зменшиться до 12 % то утруднюється дихання. В таких умовах людина напружує дихальний апарат, дихає частіше; такий стан людина витримує до 0,5 години. Перегрівання організму відбувається за умов надлишкового конвективного випромінювання тепла нагрітих поверхонь. При перегріванні вступають в активну реакцію пристосувальні функції організму. При цьому активізується робота серцево-судинної та дихальної систем, відбувається інтенсивне потовиділення, котре сягає 5 л за зміну. З потом втрачається велика кількість мінеральних солей та вітамінів. Вологість повітря суттєво впливає на терморегуляцію людського організму. Підвищення відносної вологості повітря у виробничому приміщенні ускладнює терморегуляцію, зменшення тепловиділення організмом. Фізіологічно оптимальною є відносна вологість в межах 40..60 %.

Сукупність описаних показників стану навколишнього середовища називають мікрокліматом. Для приміщень з ЕОМ встановлені норми мікроклімату приведені у таблиці 5.2.

Робота комп'ютерів і периферійних пристроїв можлива завдяки електричному струмові, до джерел якого вони підключаються, тому існує небезпека ураження користувача струмом. Щоб уникнути нещасних випадків варто строго дотримуватися правил електробезпеки.

Таблиця 5.2 – Норми мікроклімату для приміщень з ЕОМ

Пора року	Категорія робіт	Температура повітря, С, не більше	Відносна вологість повітря, %	Швидкість руху повітря, м/с
Холодна	Легка – 1а	22-24	40-60	0,1
	Легка – 1б	21-23	40-60	0,1
Тепла	Легка – 1а	23-25	40-60	0,1
	Легка – 1б	22-24	40-60	0,2

Небезпека електричного струму на відміну від інших небезпек збільшується тим, що людина не в змозі без спеціальних приладів виявити напругу дистанційно, а також швидкоплинністю поразки – небезпека виявляється, коли людина вже уражена. Ураження струмом приводить до різних порушень в організмі, викликаючи як місцеву поразку тканин і органів, так і загальне ураження організму.

Людина відчуває дратівну дію змінного струму промислової частоти силою 0,6 -1,5 мА і постійного струму 5-7 мА. Ці струми не представляють серйозної небезпеки для організму людини, тому що при такій силі струму можливе самостійне звільнення людини від контакту зі струмоведучими частинами. Для перемінного струму промислової частоти сила невідпускаючого струму знаходиться в межах 6-20 мА. Постійний струм не викликає невідпускаючого ефекту, але приводить до сильних болючих відчуттів, сила такого струму 15-80 мА і більше.

5.3 Розрахунок системи пожежогасіння офісного приміщення

Група приміщення: 1

Тип зрошувача: вода

Розміри приміщення: довжина $a = 7$ м, ширина $b = 6$ м.

1. Знайдемо групу приміщення (табл. 653) згідно з пожежним навантаженням, які забезпечуються автоматичними установками пожежогасіння (ДВН В.2.5-13-98).

За таблицею 5.3 приміщення офісу належить до 1 групи.

2. Параметри для розрахунку спринклерної установки залежно від групи приміщення (1) є наступними:

- Інтенсивність зрошування водою $L = 0,08$ л/(см²).
- Площа, яка захищається одним зрошувачем $S_{зр} = 12$ м².
- Тривалість роботи установки водного пожежогасіння $T = 30$ хвилин
- Відстань між зрошувачами $D = 4$ м.

3. Знаходимо площу приміщення:

$$S_{\text{прим}} = a * b = 7 * 6 = 42 \text{ м}^2.$$

4. Знаходимо необхідну кількість зрошувачів:

$$N = S_{\text{прим}}/S_{\text{зр}} = 42 / 12 = 3,5.$$

5. Розміщуємо зрошувачі на плані приміщення.

6. Знаходимо необхідну інтенсивність води в трубопроводі:

$$L_{\text{тр}} = L * S_{\text{прим}} = 0,08 * 42 = 3,36 \text{ л/с.}$$

7. Знаходимо інтенсивність води крізь один спринклер:

$$L_{\text{др}} = L_{\text{тр}} / N = 3,36 / 3,5 = 0,97 \text{ л/с.}$$

8. Знаходимо необхідний об'єм води:

$$Q_{\text{н}} = L_{\text{тр}} * T = 3,36 * 1800 \text{ с} = 6048 \text{ л.}$$

Таблиця 5.3 – Групи приміщень

Група	Приміщення	Пожежне навантаження
1	Приміщення книгосховищ, ЕОМ, магазинів, будівель управлінь, готелів, лікарень	до 200 Мдж/м ²
2	Приміщення з використанням рідин, які легко спалахують (ЛСР) і горять (ГР); деревообробні, швейні та інші приміщення; підприємства з обслуговування автомобілів	200...2000 Мдж/м ²
3	Приміщення гумотехнічного виробництва	200...2000 Мдж/м ²
4	Приміщення для виробництва, обробки, переробки різних матеріалів з використанням ЛСР і ГР	>2000 Мдж/м ²
5	Склади негорючих матеріалів в упаковці, що горить	>2000 Мдж/м ²
6	Склади твердих матеріалів, що горять	>2000 Мдж/м ²
7	Склади лаків, фарб, ЛСР, ГР, пластмас та ін.	>2000 Мдж/м ²

5.4 Розробка заходів щодо зменшення впливу шкідливих та небезпечних факторів

Сьогодні фахівці в області ергономіки вже зрозуміли, що не можна знайти ідеальне положення, у якому можна перебувати і працювати протягом усього робочого дня. Для більшості людей комфортабельним робочим місцем повинне бути таке, котре можна пристосувати не менш чим для двох позицій, при цьому положення крісла, монітора повинні щораз відповідати виконуваній роботі і звичкам. Багато хто вважають, що для роботи на комп'ютері більше підходять вертикальне і злегка похиле положення. Можливо, щоб крісло було злегка нахилене вперед.

Схеми розміщення робочих місць із персональним комп'ютером повинні ураховувати відстані між робочими столами з відеомоніторами (в напрямку тилу поверхні одного відеомонітора й екрана іншого відеомонітора), які повинні бути не менш 2 м, а відстань між бічними поверхнями відеомоніторів – не менш 1,2 м.

Робочий стіл повинен мати простір для ніг висотою не менш 600 мм, шириною не менш 500 мм, глибиною на рівні колін не менш 450 мм і на рівні витягнутих ніг не менш 650 мм.

Дисплей. Положення тіла звичайно відповідає напрямку погляду; дисплеї, розташовані занадто низько чи під неправильним кутом, є основними причинами сутулості. Відстань від дисплея до очей може варіюватися в залежності від характеру виконуваної роботи – 40-70 см. При роботі з текстом відстань від екрана до очей повинна лише небагато перевищувати відстань між

книгою й очима і складати 40-45 см. Необхідно давати відпочинок очам і час від часу їх просто закривати. Не можна допускати, щоб очі увесь час були сфокусовані на одній відстані, працюючи з текстом, рекомендується установлювати великий шрифт. Дуже важливо, щоб екран монітора не мерехтів. Частота регенерації зображення повинна бути не менш 72 Гц, тому що при більш низькій частоті помітне мерехтіння, хоча люди з особливо чутливим зором можуть помітити його і при більш високій частоті. Їм доведеться підібрати графічну плату і монітор з частотою регенерації 85 Гц і вище.

Необхідно уникати того, щоб монітор був звернений убік вікна, оскільки інтенсивна освітленість області зору може затопити потоками світла очі і розмити зображення на сітківці. Якщо приходить сидіти поруч з вікном, то треба розташуватися під прямим кутом до нього, причому екран дисплея повинний бути перпендикулярний шибці – цим виключаються відблиски на екрані.

Для зниження впливу низькочастотного електромагнітного випромінювання монітора на ЕЛТ рекомендується вибирати монітор, що задовольняє стандарту MPR II, чи TCO 92-95. У протилежному випадку необхідно установити на екран захисний фільтр, найбільш сучасні моделі, затримують до 99 % електромагнітного випромінювання.

Форма спинки крісла повинна повторювати форму спини працюючого. Крісло необхідно установити на такій висоті, щоб не почувався тиск на куприк (крісло розташоване занадто низько) на стегна (крісло розташоване занадто високо). Фахівці з ергономіки, вважали, що кут між стегнами і хребтом повинний складати 90°, однак недавно проведені дослідження показали, що більшість людей переважно сидять трохи відкинувшись.

Клавіатуру необхідно установити так, щоб не треба було до неї тягтися і нахилитися вперед. При зміні положення тіла (наприклад, з вертикального на похиле) обов'язково треба перемінити положення клавіатури і дисплея. Може виявитися корисною регульована підставка клавіатури, завдяки якій можна без усякої напруги працювати з маніпулятором "миша". Можна поставити клавіатуру і на коліна. Руки при роботі повинні бути прямими в зап'ястях і зігнуті в ліктях приблизно під прямим кутом. Пальці також повинні бути злегка зігнуті. Удари по клавішах не повинні бути занадто сильними.

Зручна висота столу особливо важлива в тому випадку, коли на ньому розташовується клавіатура. Якщо в клавіатури немає підставки, а висоту столу не можна змінити (і він занадто високий), то треба вище підняти сидіння крісла, а під ноги підставити лавочку: чи що-небудь інше. Якщо стіл занадто низький, необхідно підкласти що-небудь під його ніжки.

Щогодини необхідно робити перерву в роботі. У цей час рекомендується робити вправи для зап'ясть. Нижче описаний можливий комплекс вправ.

- Покласти руку на край столу долонею вниз. Взявшись, за пальці, іншою рукою відвести кисть назад і утримувати протягом 5 секунд. Повторити для іншої руки.

- Злегка упертися рукою в стіл і на 5 секунд напружити пальці в зап'ястя. Повторити іншою рукою.

- Сильно зжати пальці, а потім розпрямити їх.

Виконання приведених рекомендацій дозволяє скоротити вплив шкідливих виробничих факторів на організм людини.

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», 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 $COH \cdot 2HO$; суспензія акуратно перемішується 12 годин для повторного осадження у вигляді оксалата гадолінія.

Оксалат гадолінія $Gd(CO) \cdot 10HO$ 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 $ph = 12,6$ 000 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 картриджів 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 розміру ПЗ ІС з кодом на РНР за допомогою нелінійної регресійної моделі.

1) Для оцінювання розміру ПЗ ІС з кодом на РНР існують як лінійне так і нелінійне регресійні рівняння, які побудовано на основі метрик діаграми класів. Існуюче нелінійне регресійне рівняння для оцінювання розміру ПЗ ІС з кодом на РНР має значно кращі характеристики передбачення порівняно з лінійними рівняннями, але для нелінійного рівняння значення $PRED(0,25)$ складає 0,7188, що є меншим за 0,75, яке вважається мінімальним прийнятним значенням. Крім того регресійне рівняння на відміну від регресійної моделі не містить у своєму складі випадкової величини, яка описує похибки. А при оцінюванні розміру ПЗ як правило припускаються випадкових помилок. Тому виникає необхідність у подальшому удосконаленні нелінійного регресійного рівняння для оцінювання розміру ПЗ ІС систем з кодом на РНР.

2) Побудову нелінійної регресійної моделі для оцінювання розміру ПЗ ІС з кодом на РНР краще здійснювати за методом, що заснований на багатовимірних нормалізуючих перетвореннях та інтервалах прогнозування. Тому удосконалювати регресійну модель для оцінювання розміру ПЗ ІС з кодом на РНР доцільно саме на основі зазначеного методу.

3) Удосконалено НРМ для оцінювання розміру ПЗ ІС з кодом на РНР на основі нормалізуючого перетворення Джонсона сімейства S_B із врахуванням викидів у регресії. Ця НРМ у порівнянні з існуючими регресійними рівняннями, як лінійними так і нелінійними, що побудовано без врахування 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 факторів.

4) На підставі запропонованої нелінійної регресійної моделі розроблено програму для оцінювання розміру ПЗ ІС з кодом на РНР 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 розміру ПЗ ІС з кодом на РНР для системи моделювання Scilab 6.0.0.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Boehm B.W. Software engineering economics [Text] / B. W. Boehm. – Englewood Cliffs, NJ : Prentice Hall, 1981. – 768 p. – ISBN 0-13-822122-7.
2. Boehm B.W. Software cost estimation with COCOMO II / [B. W. Boehm, C. Abts, A. W. Brown et al.]. – Upper Saddle River, NJ: Prentice Hall PTR, 2000. – 506 p.
3. Trendowicz A. Software project effort estimation. foundations and best practice guidelines for success / A. Trendowicz, R. Jeffery. – Springer International Publishing, 2014. – 491 p. DOI: <https://doi.org/10.1007/978-3-319-03629-8>.
4. Kaczmarek J. Size and effort estimation for applications written in Java / J. Kaczmarek, M. Kucharski // Information and Software Technology. – 2004. – Vol. 46, Issue 9. – P. 589-601. DOI: <https://doi.org/10.1016/j.infsof.2003.11.001>
5. Tan H.B.K. Estimating LOC for information systems from their conceptual data models / H. B. K. Tan, Y. Zhao, H. Zhang // Proceedings of the 28th International Conference on Software Engineering (ICSE '06), Shanghai, China, May 20-28, 2006. – P. 321-330.
6. Tan H.B.K. Conceptual data model-based software size estimation for information systems / H. B. K. Tan, Y. Zhao, H. Zhang // Transactions on Software Engineering and Methodology. – 2009. – Vol. 19. – Issue 2. – October 2009. – Article No. 4.
7. Zifen Y. An improved software size estimation method based on object-oriented approach / Y. Zifen // Electrical & Electronics Engineering : IEEE Symposium EESYM'12, Kuala Lumpur, Malaysia, 24-27 June, 2012 : proceedings. – IEEE, 2012. – P. 615-617. DOI: <https://doi.org/10.1109/EEESym.2012.6258733>.

8. Kiewkanya M. Constructing C++ software size estimation model from class diagram / M. Kiewkanya, S. Surak // Computer Science and Software Engineering : 13th International Joint Conference, Khon Kaen, Thailand, July 13-15, 2016 : proceedings. – IEEE, 2016. – P. 1-6. DOI: <https://doi.org/10.1109/JCSSE.2016.7748880>

9. Prykhodko S.B. Constructing the non-linear regression equation to estimate the software size of open source PHP-based information systems / [S. B. Prykhodko, N. V. Prykhodko, T. G. Smykodub et al.] // Problems of Information Technology. – 2018. – № 1 (023). – P. 118-125. – ISSN 1998-7005

10. Prykhodko N.V. The non-linear regression model to estimate the software size of open source Java-based systems / N. V. Prykhodko, S. B. Prykhodko // Radio Electronics, Computer Science, Control. – 2018. – No. 3 (46). – P. 158-166. DOI: <https://doi.org/10.15588/1607-3274-2018-3-17>

11. Приходько Н.В. Нелінійна регресійна модель для оцінювання розміру програмного забезпечення інформаційних систем на базі VB / Н. В. Приходько, С. Б. Приходько // Інформаційні технології та комп'ютерна інженерія. – 2018. – Т.43. – № 3. – С. 37-42. – ISSN 1999-9941. DOI: <https://doi.org/10.31649/1999-9941-2018-43-3-37-42>

12. Приходько Н.В. Нелінійна регресійна модель для оцінювання розміру програмного забезпечення промислових інформаційних систем на Java / Н. В. Приходько, С. Б. Приходько // Моделювання та інформаційні технології : зб. наук. пр. / Нац. акад. наук України, Ін-т пробл. моделювання в енергетиці ім. Г. Є. Пухова. – Київ: [б. в.], 2018. – Вип. 85. – С. 81-88. ISSN 2309-7647 (print).

13. Prykhodko S. Estimating the Software Size of Open-Source PHP-Based Systems Using Non-Linear Regression Analysis / Sergiy Prykhodko, Natalia Prykhodko, Lidiia Makarova // Proceedings of International Conference

“Advanced Computer Information Technologies” (ACIT-2018). CEUR Workshop Proceedings. – 2019. – Vol.2300. – Ceske Budejovice, CZECH REPUBLIC. CEUR-WS.org – P. 199-202. ISSN 1613-0073 Access mode: <http://ceur-ws.org/Vol-2300/Paper48.pdf>

14. Приходько С.Б. Трьохфакторна нелінійна регресійна модель для оцінювання розміру Php-застосунків з відкритим кодом / [С. Б. Приходько, Н. В. Приходько, Т. А. Фаріонова та ін.] // Науковий журнал «Вчені записки Таврійського національного університету імені В. І. Вернадського. Серія: Технічні науки» Том 31 (70) № 1, 2020. – С. 124-131. DOI: <https://doi.org/10.32838/2663-5941/2020.1-1/23>

15. Приходько С.Б. Чотирьохфакторна нелінійна регресійна модель для оцінювання розміру Java-застосунків з відкритим кодом / С. Б. Приходько, Н. В. Приходько, Т. Г. Смикодуб // Науковий журнал «Вчені записки Таврійського національного університету імені В. І. Вернадського. Серія: Технічні науки» Том 31 (70) № 2, 2020. – С. 157-162. DOI: <https://doi.org/10.32838/2663-5941/2020.2-1/25>

16. Приходько С.Б. Нелінійна регресійна модель для оцінювання розміру Web-застосунків, що створюються з використанням фреймворку Laravel / [С. Б. Приходько, Н. В. Приходько, М. В. Ворона та ін.] // Інформаційні технології та комп'ютерна інженерія. – 2021. – Т.50. – № 1. – С. 115-121. – ISSN: 1999-9941. DOI: <https://doi.org/10.31649/1999-9941-2021-50-1-115-121>

17. Приходько С.Б. Оцінювання розміру PHP-застосунків з відкритим кодом за нелінійними регресійними моделями з різними факторами / С. Б. Приходько, М. В. Ворона // Збірник наукових праць Національного університету кораблебудування імені адмірала Макарова. – 2021. – №1. – С. 92-98. ISSN: 2311-3405. DOI: [https://doi.org/10.15589/znp2021.1\(484\).13](https://doi.org/10.15589/znp2021.1(484).13)

18. Daud M. Improving the accuracy of early software size estimation using analysis-to-design adjustment factors (ADAFs) / M. Daud, A. A. Malik // *IEEE Access*. – 2021. – Vol. 9. – P. 81986-81999. DOI: <https://doi.org/10.1109/ACCESS.2021.3085752>

19. Zhang K. Efficiency improvement of function point-based software size estimation with deep learning model / K. Zhang, X. Wang, J. Ren, and C. Liu // *IEEE Access*. – 2021. – Vol. 9. – P. 107124-107136. DOI: <https://doi.org/10.1109/ACCESS.2020.2998581>

20. Manisha. Early size estimation using machine learning / Manisha, R. Rishi // 2021 8th International Conference on Computing for Sustainable Global Development (INDIACom): proceedings. – 2021. – P. 757-762. DOI: <https://doi.org/10.1109/INDIACom51348.2021.00135>

21. 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>

22. Приходько С.Б. Метод покращення нелінійних регресійних моделей на основі багатовимірних нормалізуючих перетворень / С. Б. Приходько, Н. В. Приходько // Прикладні науково-технічні дослідження: матеріали III міжнар. наук.-практ. конф., 3-5 квіт. 2019 р. – Академія технічних наук України. – Івано-Франківськ: Симфонія форте, 2019. – С. 20. – ISBN 978-966-284-132-9

23. 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

24. Приходько С.Б. Нелінійна регресійна модель для оцінювання розміру програмного забезпечення інформаційних систем з кодом на PHP / С. Б. Приходько, **Г. В. Сербулов** // Інформаційні технології: моделі, алгоритми, системи (ITMAS – 2021): Матеріали II-ої Всеукраїнської науково-практичної інтернет конференції (26-28 жовтня 2021 р.). – Миколаїв: НУК імені адмірала Макарова, 2021. – С. 13-14. Режим доступу: <http://itconf.nuos.edu.ua/2021/proceedings/>

25. 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.

26. 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.

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

28. 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.

29. 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.

Додаток А – Технічне завдання на розробку програми для оцінювання розміру програмного забезпечення інформаційних систем з кодом на РНР

Вступ

Назва розробки – “Програма для оцінювання розміру програмного забезпечення інформаційних систем з кодом на РНР”, далі програма.

Підстави для розробки

Завдання на кваліфікаційну роботу.

Призначення розробки:

Програма призначена для оцінювання розміру ПЗ інформаційних систем з кодом на РНР 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 ІС з кодом на РНР: перша метрика Y включає фактичний розмір ПЗ в тисячах строк коду (KLOC), друга X_1 та третя X_2 метрики визначають відповідно загальну кількість класів і загальну кількість співвідношень у концептуальній моделі даних у вигляді діаграми класів, а останньою четвертою метрикою X_3 є середня кількість атрибутів на клас, тобто відношення загальної кількості атрибутів у концептуальній моделі даних до X_1 .

2) Вводити з інтерфейсу програми довірчу ймовірність, три метрики концептуальної моделі даних у вигляді діаграми класів, за якими потрібно оцінити розмір ПЗ ІС з кодом на РНР в тисячах строк коду: загальну кількість

класів X_1 і загальну кількість співвідношень X_2 у концептуальній моделі даних у вигляді діаграми класів, середню кількість атрибутів на клас X_3 , тобто відношення загальної кількості атрибутів у концептуальній моделі даних до X_1 .

3) Виконувати нормалізацію емпіричних даних за нормалізуючим перетворенням Джонсона сімейства S_B .

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

5) Здійснювати побудову НРМ.

6) Обчислювати значення коефіцієнту детермінації R^2 .

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

8) Виводити на екран нелінійну регресію, довірчий інтервал та інтервал прогнозування нелінійної регресії розміру ПЗ ІС з кодом на РНР в тисячах строк коду, значення коефіцієнту детермінації R^2 .

9) Виводити у текстовий файл результати оцінювання розміру ПЗ ІС з кодом на РНР в тисячах строк коду.

Вимоги до організації вхідних та вихідних даних

Вхідні дані:

- емпіричні дані з метрик ПЗ ІС з кодом на РНР: перша метрика Y включає фактичний розмір ПЗ в тисячах строк коду (KLOC), друга X_1 та третя X_2 метрики визначають відповідно загальну кількість класів і загальну кількість співвідношень у концептуальній моделі даних, а останньою четвертою метрикою X_3 є середня кількість атрибутів на клас, тобто відношення загальної кількості атрибутів у концептуальній моделі даних до X_1 ;

- значення довірчої ймовірності, трьох метрик концептуальної моделі даних, за якими потрібно оцінити розмір ПЗ ІС з кодом на РНР (в тисячах строк коду): загальну кількість класів X_1 і загальну кількість співвідношень X_2 у концептуальній моделі даних, середню кількість атрибутів на клас X_3 , тобто відношення загальної кількості атрибутів у концептуальній моделі даних до X_1 .

Вихідні дані:

- значення нелінійної регресії, границь довірчого інтервалу та інтервалу прогнозування нелінійної регресії розміру ПЗ ІС з кодом на РНР.

- значення коефіцієнту детермінації.

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 С (Visual Studio 2010 або Visual Express 2010) для компіляції зовнішніх модулів С або С++ 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	21.09.2021
2. Ескізний проект	Розробка ескізного проекту	24.09.2021
	Затвердження ескізного проекту	26.09.2021
3. Технічний проект	Розробка технічного проекту	08.10.2021
	Затвердження технічного проекту	10.10.2021
4. Робочий проект	Розробка програми	05.11.2021
	Розробка програмної документації	12.11.2021
	Випробування програми	21.11.2021

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 розміру ПЗ інформаційних систем з кодом на PHP
f=figure();//Создание графического окна.
//Встановлюємо розміри вікна.
set(f,'position',[0,0,700,300])
//Встановлюємо заголовок вікна.
set(f,'figure_name',...
'Oцінювання розміру ПЗ інформаційних систем з кодом на PHP');

//plot2d(x,y,-9);

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\PHP_IS_30.txt','position',[320,240,140,20]);

BtDatInp=uicontrol('style','pushbutton','string','Ввести',...
'CallBack','[x,yx,f1,f2,f3,n]=DataInput()','position',[340,210,100,20]);

//lab_text1=uicontrol(f,'style','text','string',...
//'Параметри нормалізуючого перетворення Джонсона','position',...
//[100,270,300,20]);
lab_text2=uicontrol(f,'style','text','string',...
'Наявність фактичного розміру ПЗ','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',...
'Загальна кількість класів','position',...
[20,240,200,20]);
edit_text4=uicontrol(f,'style','edit','string',...
'5','position',[230,240,60,20]);
lab_text5=uicontrol(f,'style','text','string',...
'Загальна кількість співвідношень','position',...
[20,210,200,20]);
edit_text5=uicontrol(f,'style','edit','string',...
'2','position',[230,210,60,20]);
lab_text6=uicontrol(f,'style','text','string',...
'Середня кількість атрибутів на клас','position',...
[20,180,200,20]);
edit_text6=uicontrol(f,'style','edit','string',...
'10.6','position',[230,180,60,20]);
lab_text7=uicontrol(f,'style','text','string',...
'Довірча ймовірність','position',...
[20,150,200,20]);
edit_text7=uicontrol(f,'style','edit','string',...
'0.95','position',[230,150,60,20]);
lab_text8=uicontrol(f,'style','text','string',...
'Результати оцінювання','position',...
```

```

[50,120,120,20]);
lab_text9=uicontrol(f,'style','text','string',...
'Reзультат нел. регресії розміру ПЗ (KLOC)','position',...
[20,90,245,20]);
lab_text10=uicontrol(f,'style','text','string',...
'Нижня границя довір.інтервалу розміру (KLOC)','position',...
[20,60,245,20]);
lab_text11=uicontrol(f,'style','text','string',...
'Верхня границя довір.інтервалу розміру (KLOC)','position',...
[20,30,245,20]);
lab_text12=uicontrol(f,'style','text','string',...
'Нижня границя інтервалу прогноз. розміру (KLOC)','position',...
[355,60,260,20]);
lab_text13=uicontrol(f,'style','text','string',...
'Верхня границя інтервалу прогноз. розміру (KLOC)','position',...
[355,30,260,20]);
//
//Текстове поле, що визначає виведення результатів.
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]);

//Прапорець, який відповідає за вибір.
radio_bikv_x=uicontrol('style','radiobutton','string',...
'є (включити), немає (виключити)', 'value',1, 'position',...
[490,240,200,20]);
//
//radio_bikv_y=uicontrol('style','radiobutton','string',...
//'SU (включити), SB (виключити)', 'value',1, 'position',...
//[500,180,300,20]);
//
//Перевіряємо значення прапорця, якщо прапорець вимкнено,
//if (get(radio_bikv_x,'value')==0) & (get(radio_bikv_y,'value')==0)
if (get(radio_bikv_x,'value')==0)

else
//      set(textresult,'string',...
//      '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',...
.
'PHPSoftSizeEst(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, f3, n]=DataInput()
dir_name=get(edit_text,'string');

file_PC=mopen(dir_name,'r');
file_PC2=mopen('D:\Data_F\PHP_IS_30_data2.txt','w+');
file_PC3=mopen('D:\Data_F\PHP_IS_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'); x3(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); yx(i,4)=x3(i);
    x(i,1)=x1(i); x(i,2)=x2(i); x(i,3)=x3(i);
end;
//for i=1:n
    fprintf(file_PC2,'%f\t%f\t%f\t%f\n',y,x1,x2,x3);
//end;

for i = 1:n
    for j = 1:3 fprintf(file_PC3, "%5.3f ", x(i,j)); end;
    fprintf(file_PC3, "\n");
end;

fclose(file_PC2);
fclose(file_PC3);
f1=figure();//Створення графічного вікна.
//Встановлюємо розміри вікна.
set(f1,'position',[50,50,700,300]);
//Встановлюємо заголовок вікна.
set(f1,'figure_name','Емпіричні дані: розмір (у тисячах строк коду) в залежності від загальної кількості класів');
plot2d(x1,y,-9);

f2=figure();//Створення графічного вікна.
//Встановлюємо розміри вікна.
set(f2,'position',[50,50,700,300]);
//Встановлюємо заголовок вікна.
set(f2,'figure_name','Емпіричні дані: розмір (у тисячах строк коду) в залежності від загальної кількості співвідношень');
plot2d(x2,y,-9);

f3=figure();//Створення графічного вікна.
//Встановлюємо розміри вікна.

```

```

set(f3,'position',[50,50,700,300]);
//Встановлюємо заголовок вікна.
set(f3,'figure_name','Емпіричні дані: розмір (у тисячах строк коду) в залежності
від середньої кількості атрибутів на клас');
plot2d(x3,y,-9);

x1A=0; x2A=0; x3A=0;
for i=1:n
    x1A=x1A+x(i,1); x2A=x2A+x(i,2); x3A=x3A+x(i,3);
end;
x1A=x1A/n; x2A=x2A/n; x3A=x3A/n;
Sx1x1=0; Sx2x2=0; Sx3x3=0;
Sx1x2=0; Sx1x3=0; Sx2x3=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);
    Sx1x3=Sx1x3+(x(i,1)-x1A)*(x(i,3)-x3A);
    Sx2x3=Sx2x3+(x(i,2)-x2A)*(x(i,3)-x3A);
    Sx2x2=Sx2x2+(x(i,2)-x2A)*(x(i,2)-x2A);
    Sx3x3=Sx3x3+(x(i,3)-x3A)*(x(i,3)-x3A);
end;
Sxx(1,1)=Sx1x1; Sxx(1,2)=Sx1x2; Sxx(1,3)=Sx1x3;
Sxx(2,1)=Sx1x2; Sxx(2,2)=Sx2x2; Sxx(2,3)=Sx2x3;
Sxx(3,1)=Sx1x3; Sxx(3,2)=Sx2x3; Sxx(3,3)=Sx3x3;

Sxx_1=1/Sxx;

ay=1.091752366476;
ax1=1.007190; ax2=0.812284514764; ax3=0.7896157118;
by=0.60232130;
bx1=0.74733212763; bx2=0.77582060; bx3=0.581457617658;
cy=1.602;
cx1=2.08914350; cx2=0.247154038; cx3=3.233;
dy=101.3795327;
dx1=48.0019355; dx2=28.086461764; dx3=14.249170;

z1A=0; z2A=0; z3A=0;
for i=1:n
    //z1A=z1A+log10(x(i,1)); z2A=z2A+log10(x(i,2)); z3A=z3A+log10(x(i,3));
    z1A=z1A+ax1+bx1*log((x(i,1)-cx1)/(dx1-x(i,1)+cx1));
    z2A=z2A+ax2+bx2*log((x(i,2)-cx2)/(dx2-x(i,2)+cx2));
    z3A=z3A+ax3+bx3*log((x(i,3)-cx3)/(dx3-x(i,3)+cx3));
end;
z1A=z1A/n; z2A=z2A/n; z3A=z3A/n;
Sz1z1=0; Sz2z2=0; Sz3z3=0;
Sz1z2=0; Sz1z3=0; Sz2z3=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);
    //Sz1z3=Sz1z3+(log10(x(i,1))-z1A)*(log10(x(i,3))-z3A);
    //Sz2z3=Sz2z3+(log10(x(i,2))-z2A)*(log10(x(i,3))-z3A);
    //Sz2z2=Sz2z2+(log10(x(i,2))-z2A)*(log10(x(i,2))-z2A);
    //Sz3z3=Sz3z3+(log10(x(i,3))-z3A)*(log10(x(i,3))-z3A);

```

```

    Sz1z1=Sz1z1+(ax1+bx1*log((x(i,1)-cx1)/(dx1-x(i,1)+cx1))-
z1A)*(ax1+bx1*log((x(i,1)-cx1)/(dx1-x(i,1)+cx1))-z1A);
    Sz1z2=Sz1z2+(ax1+bx1*log((x(i,1)-cx1)/(dx1-x(i,1)+cx1))-
z1A)*(ax2+bx2*log((x(i,2)-cx2)/(dx2-x(i,2)+cx2))-z2A);
    Sz1z3=Sz1z3+(ax1+bx1*log((x(i,1)-cx1)/(dx1-x(i,1)+cx1))-
z1A)*(ax3+bx3*log((x(i,3)-cx3)/(dx3-x(i,3)+cx3))-z3A);
    Sz2z3=Sz2z3+(ax2+bx2*log((x(i,2)-cx2)/(dx2-x(i,2)+cx2))-
z2A)*(ax3+bx3*log((x(i,3)-cx3)/(dx3-x(i,3)+cx3))-z3A);
    Sz2z2=Sz2z2+(ax2+bx2*log((x(i,2)-cx2)/(dx2-x(i,2)+cx2))-
z2A)*(ax2+bx2*log((x(i,2)-cx2)/(dx2-x(i,2)+cx2))-z2A);
    Sz3z3=Sz3z3+(ax3+bx3*log((x(i,3)-cx3)/(dx3-x(i,3)+cx3))-
z3A)*(ax3+bx3*log((x(i,3)-cx3)/(dx3-x(i,3)+cx3))-z3A);
end;
Szz(1,1)=Sz1z1; Szz(1,2)=Sz1z2; Szz(1,3)=Sz1z3;
Szz(2,1)=Sz1z2; Szz(2,2)=Sz2z2; Szz(2,3)=Sz2z3;
Szz(3,1)=Sz1z3; Szz(3,2)=Sz2z3; Szz(3,3)=Sz3z3;

Szz_1=1/Szz;

//zi(1)=log10(x(1,1))-z1A; zi(2)=log10(x(1,2))-z2A; zi(3)=log10(x(1,3))-z3A;
zi(1)=ax1+bx1*log((x(i,1)-cx1)/(dx1-x(i,1)+cx1))-z1A;
zi(2)=ax2+bx2*log((x(i,2)-cx2)/(dx2-x(i,2)+cx2))-z2A;
zi(3)=ax3+bx3*log((x(i,3)-cx3)/(dx3-x(i,3)+cx3))-z3A;

ziT=zi';
ziSzz_1=ziT*Szz_1;
result=ziSzz_1*zi;

Df=n-4;
P=0.975;
Q=1-P;

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

//    Delta_z=T/n^0.5;

b0=0;
b1=0.47275475;
b2=0.5262282833;
b3=0.07203592772;

Szyzyn4=0;
for i=1:n
    z_y=ay+by*log((y(x(i,1))-cy)/(dy-y(x(i,1))+cy));
    z_x1=ax1+bx1*log((x(i,1)-cx1)/(dx1-x(i,1)+cx1));
    z_x2=ax2+bx2*log((x(i,2)-cx2)/(dx2-x(i,2)+cx2));
    z_x3=ax3+bx3*log((x(i,3)-cx3)/(dx3-x(i,3)+cx3));
    Szyzyn4=Szyzyn4+(b0+b1*z_x1+b2*z_x2+b3*z_x3-z_y)^2;
    //Szyzyn4=Szyzyn4+...
    //(b0+b1*log10(x(i,1))+b2*log10(x(i,2))+b3*log10(x(i,3))-log10(y(x(i,1))))^2;
end;
Szyzyn4=Szyzyn4/(n-4)
Delta_z=T*(Szyzyn4*(1/n+result))^0.5;

```

```

z_x1=ax1+bx1*log((x(1,1)-cx1)/(dx1-x(1,1)+cx1));
z_x2=ax2+bx2*log((x(1,2)-cx2)/(dx2-x(1,2)+cx2));
z_x3=ax3+bx3*log((x(1,3)-cx3)/(dx3-x(1,3)+cx3));
zy_result=b0+b1*z_x1+b2*z_x2+b3*z_x3;
//zy_result=b0+b1*log10(x(1,1))+b2*log10(x(1,2))+b3*log10(x(1,3));

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

y_result=cy+dy/(1+exp(-(zy_result-ay)/by));
yLBC=cy+dy/(1+exp(-(zyLBC-ay)/by));
yUBC=cy+dy/(1+exp(-(zyUBC-ay)/by));

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

file_PC5=fopen('D:\Data_F\PHP_IS_30_data5.txt','w+');
//for i=1:3
    fprintf(file_PC5, "%5.6f ", yUBC);
//end
fclose(file_PC5);

file_PC4=fopen('D:\Data_F\PHP_IS_30_data4.txt','w+');
for i = 1:3
    for j = 1:3 fprintf(file_PC4, "%5.6f ", Szz_1(i,j)); end;
    fprintf(file_PC4, "\n");
end;
fclose(file_PC4);

endfunction

//Функція оцінювання розміру ПЗ інформаційних систем з кодом на PHP
function PHPSoftSizeEst(x, yx, n)
// Зчитуємо значення змінних із текстових полів та
// перетворюємо їх до числового типу.
x1i=eval(get(edit_text4,'string'));
x2i=eval(get(edit_text5,'string'));
x3i=eval(get(edit_text6,'string'));
Pconf=eval(get(edit_text7,'string'));

x1A=0; x2A=0; x3A=0; yA=0;
for i=1:n
    x1A=x1A+x(i,1); x2A=x2A+x(i,2); x3A=x3A+x(i,3); yA=yA+yx(i,1);
end;
x1A=x1A/n; x2A=x2A/n; x3A=x3A/n; yA=yA/n;
Sx1x1=0; Sx2x2=0; Sx3x3=0;
Sx1x2=0; Sx1x3=0; Sx2x3=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);
    Sx1x3=Sx1x3+(x(i,1)-x1A)*(x(i,3)-x3A);
    Sx2x3=Sx2x3+(x(i,2)-x2A)*(x(i,3)-x3A);
    Sx2x2=Sx2x2+(x(i,2)-x2A)*(x(i,2)-x2A);
    Sx3x3=Sx3x3+(x(i,3)-x3A)*(x(i,3)-x3A);
    SyA=SyA+(yx(i,1)-yA)*(yx(i,1)-yA);

```

```

end;
Sxx(1,1)=Sx1x1; Sxx(1,2)=Sx1x2; Sxx(1,3)=Sx1x3;
Sxx(2,1)=Sx1x2; Sxx(2,2)=Sx2x2; Sxx(2,3)=Sx2x3;
Sxx(3,1)=Sx1x3; Sxx(3,2)=Sx2x3; Sxx(3,3)=Sx3x3;

Sxx_1=1/Sxx;

ay=1.091752366476;
ax1=1.007190; ax2=0.812284514764; ax3=0.7896157118;
by=0.60232130;
bx1=0.74733212763; bx2=0.77582060; bx3=0.581457617658;
cy=1.602;
cx1=2.08914350; cx2=0.247154038; cx3=3.233;
dy=101.3795327;
dx1=48.0019355; dx2=28.086461764; dx3=14.249170;

z1A=0; z2A=0; z3A=0;
for i=1:n
    z1A=z1A+ax1+bx1*log((x(i,1)-cx1)/(dx1-x(i,1)+cx1));
    z2A=z2A+ax2+bx2*log((x(i,2)-cx2)/(dx2-x(i,2)+cx2));
    z3A=z3A+ax3+bx3*log((x(i,3)-cx3)/(dx3-x(i,3)+cx3));
    //z1A=z1A+log10(x(i,1)); z2A=z2A+log10(x(i,2)); z3A=z3A+log10(x(i,3));
end;
z1A=z1A/n; z2A=z2A/n; z3A=z3A/n;
Sz1z1=0; Sz2z2=0; Sz3z3=0;
Sz1z2=0; Sz1z3=0; Sz2z3=0;
for i=1:n
    Sz1z1=Sz1z1+(ax1+bx1*log((x(i,1)-cx1)/(dx1-x(i,1)+cx1))-
    z1A)*(ax1+bx1*log((x(i,1)-cx1)/(dx1-x(i,1)+cx1))-z1A);
    Sz1z2=Sz1z2+(ax1+bx1*log((x(i,1)-cx1)/(dx1-x(i,1)+cx1))-
    z1A)*(ax2+bx2*log((x(i,2)-cx2)/(dx2-x(i,2)+cx2))-z2A);
    Sz1z3=Sz1z3+(ax1+bx1*log((x(i,1)-cx1)/(dx1-x(i,1)+cx1))-
    z1A)*(ax3+bx3*log((x(i,3)-cx3)/(dx3-x(i,3)+cx3))-z3A);
    Sz2z3=Sz2z3+(ax2+bx2*log((x(i,2)-cx2)/(dx2-x(i,2)+cx2))-
    z2A)*(ax3+bx3*log((x(i,3)-cx3)/(dx3-x(i,3)+cx3))-z3A);
    Sz2z2=Sz2z2+(ax2+bx2*log((x(i,2)-cx2)/(dx2-x(i,2)+cx2))-
    z2A)*(ax2+bx2*log((x(i,2)-cx2)/(dx2-x(i,2)+cx2))-z2A);
    Sz3z3=Sz3z3+(ax3+bx3*log((x(i,3)-cx3)/(dx3-x(i,3)+cx3))-
    z3A)*(ax3+bx3*log((x(i,3)-cx3)/(dx3-x(i,3)+cx3))-z3A);
    //Sz1z1=Sz1z1+(log10(x(i,1))-z1A)*(log10(x(i,1))-z1A);
    //Sz1z2=Sz1z2+(log10(x(i,1))-z1A)*(log10(x(i,2))-z2A);
    //Sz1z3=Sz1z3+(log10(x(i,1))-z1A)*(log10(x(i,3))-z3A);
    //Sz2z3=Sz2z3+(log10(x(i,2))-z2A)*(log10(x(i,3))-z3A);
    //Sz2z2=Sz2z2+(log10(x(i,2))-z2A)*(log10(x(i,2))-z2A);
    //Sz3z3=Sz3z3+(log10(x(i,3))-z3A)*(log10(x(i,3))-z3A);
end;
Szz(1,1)=Sz1z1; Szz(1,2)=Sz1z2; Szz(1,3)=Sz1z3;
Szz(2,1)=Sz1z2; Szz(2,2)=Sz2z2; Szz(2,3)=Sz2z3;
Szz(3,1)=Sz1z3; Szz(3,2)=Sz2z3; Szz(3,3)=Sz3z3;

Szz_1=1/Szz;

zi(1)=ax1+bx1*log((x1i-cx1)/(dx1-x1i+cx1))-z1A;
zi(2)=ax2+bx2*log((x2i-cx2)/(dx2-x2i+cx2))-z2A;

```

```

zi(3)=ax3+bx3*log((x3i-cx3)/(dx3-x3i+cx3))-z3A;
//zi(1)=log10(x1i)-z1A; zi(2)=log10(x2i)-z2A; zi(3)=log10(x3i)-z3A;
ziT=zi';
ziSzz_1=ziT*Szz_1;
result=ziSzz_1*zi;

Df=n-4;
alpha=1-Pconf;
P=1-alpha/2;
//P=0.975;
Q=1-P;

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

b0=0;
b1=0.47275475;
b2=0.5262282833;
b3=0.07203592772;

Szyzyn4=0; Syy=0;
for i=1:n
    z_y=ay+by*log((yx(i,1)-cy)/(dy-yx(i,1)+cy));
    z_x1=ax1+bx1*log((x(i,1)-cx1)/(dx1-x(i,1)+cx1));
    z_x2=ax2+bx2*log((x(i,2)-cx2)/(dx2-x(i,2)+cx2));
    z_x3=ax3+bx3*log((x(i,3)-cx3)/(dx3-x(i,3)+cx3));
    Szyzyn4=Szyzyn4+(b0+b1*z_x1+b2*z_x2+b3*z_x3-z_y)^2;

    //Szyzyn4=Szyzyn4+.
    //(b0+b1*log10(x(i,1))+b2*log10(x(i,2))+b3*log10(x(i,3))-log10(yx(i,1)))^2;

    zy_result=b0+b1*z_x1+b2*z_x2+b3*z_x3;
    Syy=Syy+(yx(i,1)-(cy+dy/(1+exp(-(zy_result-ay)/by))))^2;
    //Syy=Syy+(yx(i,1)-10^b0*x(i,1)^b1*x(i,2)^b2*x(i,3)^b3)^2;
end;
Szyzyn4=Szyzyn4/(n-4)
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;

z_x1=ax1+bx1*log((x(1,1)-cx1)/(dx1-x(1,1)+cx1));
z_x2=ax2+bx2*log((x(1,2)-cx2)/(dx2-x(1,2)+cx2));
z_x3=ax3+bx3*log((x(1,3)-cx3)/(dx3-x(1,3)+cx3));
zy_result=b0+b1*z_x1+b2*z_x2+b3*z_x3;

//zy_result=b0+b1*log10(x1i)+b2*log10(x2i)+b3*log10(x3i);

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

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

y_result=cy+dy/(1+exp(-(zy_result-ay)/by));
yLBC=cy+dy/(1+exp(-(zyLBC-ay)/by));
yUBC=cy+dy/(1+exp(-(zyUBC-ay)/by));

```

```

yLBPr=cy+dy/(1+exp(-(zyLBPr-ay)/by));
yUBPr=cy+dy/(1+exp(-(zyUBPr-ay)/by));

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

file_PC5=mopen('D:\Data_F\PHP_IS_30_data5.txt','w+');
//for i=1:3
    fprintf(file_PC5, "%5.6f ", yUBC);
//end
fclose(file_PC5);

file_PC4=mopen('D:\Data_F\PHP_IS_30_data4.txt','w+');
for i = 1:3
    for j = 1:3 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 радио_bikv_x,'value')==0)

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

endfunction

//Функція закриття вікон.
function W_Close()
    close(f);
    close(f1);
    close(f2);
    close(f3);
endfunction

```

ДОДАТОК В – Опис програми

Загальні відомості

Програма для оцінювання розміру програмного забезпечення інформаційних систем з кодом на PHP для системи моделювання Scilab 6.0.0. Файл програми PHPSoftSizeEst.sce має розмір 17 КБ.

Функціональне призначення

Програма PHPSoftSizeEst призначений для оцінювання розміру програмного забезпечення інформаційних систем з кодом на PHP 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 PHPSoftSizeEst.sce стандартними засобами ОС.

ДОДАТОК Г – Інструкція користувача

Загальний опис роботи з програмою

Після завантаження файлу PHPSoftSizeEst.sce на екрані з'являється форма для відображення початкового стану програми для оцінювання розміру ПЗ інформаційних систем з кодом на PHP (рисунок Г.1).

Рисунок Г.1 – Форма для відображення початкового стану програми

Далі у вікні редагування «Файл з даними» графічного вікна «Оцінювання розміру ПЗ інформаційних систем з кодом на PHP» (рис.Г.1) потрібно відредагувати ім'я текстового файлу з початковими емпіричними даними, що

будуть використовуватися для оцінювання розміру програмного забезпечення інформаційних систем з кодом на РНР.

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

Потім у графічному вікні «Оцінювання розміру ПЗ інформаційних систем з кодом на РНР» (рис. Г.1) у відповідних вікнах редагування потрібно відредагувати: загальну кількість класів і загальну кількість співвідношень у концептуальній моделі даних, середню кількість атрибутів на клас, тобто відношення загальної кількості атрибутів у концептуальній моделі даних до кількості класів, довірчу ймовірність.

Крім того у графічному вікні «Оцінювання розміру ПЗ інформаційних систем з кодом на РНР» (рис. Г.1) потрібно за допомогою радіо кнопки здійснити вибір наявності фактичного розміру ПЗ та ввести його для подальшого порівняння з відповідною оцінкою кількістю строк коду.

Далі потрібно натиснути на командну кнопку «Оцінювання». Після цього з'являться результати оцінювання розміру ПЗ інформаційної системи з кодом на РНР.

І на останок у графічному вікні «Оцінювання розміру ПЗ інформаційних систем з кодом на РНР» 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 розміру ПЗ інформаційних систем з кодом на PHP PHPSoftSizeEst.

Ціль випробувань: перевірка відповідності програми PHPSoftSizeEst 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 PHPSoftSizeEst. 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 виконаною, 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 розміру ПЗ інформаційних систем з кодом на PHP

File Tools Edit ?

Оцінювання розміру ПЗ інформаційних систем з кодом на PHP

Дані для оцінювання		Файл з даними	Наявність фактичного розміру ПЗ
Загальна кількість класів	5	D:\Data_F\PHP_IS_30.txt	☒ є (включити), немає (виключити)
Загальна кількість співвідношень	2	Ввести	0
Середня кількість атрибутів на клас	10.6		
Довірча ймовірність	0.95	Оцінювання	
Результати оцінювання			
Результат нел. регресії розміру ПЗ (KLOC)			
Нижня границя довір.інтервалу розміру (KLOC)		Нижня границя інтервалу прогноз. розміру (KLOC)	
Верхня границя довір.інтервалу розміру (KLOC)		Верхня границя інтервалу прогноз. розміру (KLOC)	
Закрити			

Рисунок Д.1 – Графічне вікно головного модуля програми

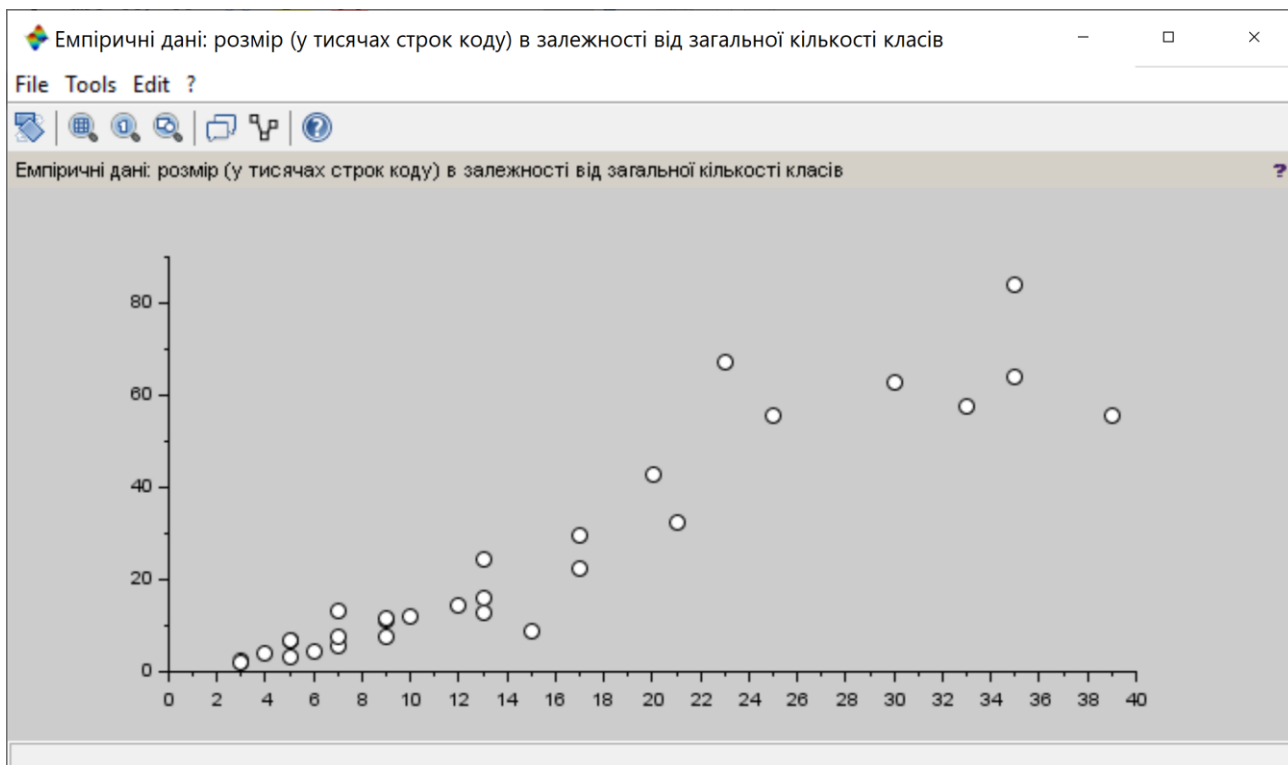


Рисунок Д.2 – Візуалізація тестових емпіричних даних.

Оцінювання розміру ПЗ інформаційних систем з кодом на PHP

File Tools Edit ?

Оцінювання розміру ПЗ інформаційних систем з кодом на PHP

Дані для оцінювання		Файл з даними	Наявність фактичного розміру ПЗ	
Загальна кількість класів	5	D:\Data_F\PHP_IS_30.txt	☐ є (включити), немає (виключити)	
Загальна кількість співвідношень	2	Ввести	0	
Середня кількість атрибутів на клас	10.6			
Довірча ймовірність	0.95	Оцінювання		
Результати оцінювання		Коефіцієнт детермінації $R^2=0.9756$		
Результат нел. регресії розміру ПЗ (KLOC)		=4.1524		
Нижня границя довір.інтервалу розміру (KLOC)		Нижня границя інтервалу прогноз. розміру (KLOC)		=2.8658
Верхня границя довір.інтервалу розміру (KLOC)		Верхня границя інтервалу прогноз. розміру (KLOC)		=6.6823
Закрити				

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

Оцінювання розміру ПЗ інформаційних систем з кодом на PHP

File Tools Edit ?

Оцінювання розміру ПЗ інформаційних систем з кодом на PHP

Дані для оцінювання		Файл з даними	Наявність фактичного розміру ПЗ
Загальна кількість класів	5	D:\Data_F\PHP_IS_30.txt	☒ є (включити), немає (виключити)
Загальна кількість співвідношень	2	Ввести	3.038
Середня кількість атрибутів на клас	10.6		
Довірча ймовірність	0.95	Оцінювання	
Результати оцінювання		Коефіцієнт детермінації $R^2=0.9756$	
Результат нел. регресії розміру ПЗ (KLOC)		Відносна похибка MRE=0.3668	
Нижня границя довір.інтервалу розміру (KLOC)		Нижня границя інтервалу прогноз. розміру (KLOC)	
Верхня границя довір.інтервалу розміру (KLOC)		Верхня границя інтервалу прогноз. розміру (KLOC)	

Закрити

Рисунок Д.4 – Результати роботи програми після натискання на командну кнопку «Оцінювання» у разі, коли радіо кнопка включена та введений фактичний розмір ПЗ – 3,038 тис. строк коду

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

```

3.038 5 2 10.6
22.599 17 7 7
32.243 21 13 4.524
16.164 13 11 7.077
83.862 35 24 6.571
24.22 13 9 8.077
63.929 35 19 8.029
6.697 5 5 7
55.537 25 14 8.64
55.752 39 10 9.077

```

62.602 30 17 7
67.111 23 22 14.957
2.552 3 1 8.333
12.17 10 5 3.7
12.757 13 9 5
5.695 7 3 8.429
7.744 9 6 9.222
11.054 9 9 3.667
29.77 17 15 3.412
11.653 9 8 8.778
6.847 5 4 3.6
13.389 7 5 11.714
14.45 12 6 16.583
4.414 6 3 3.667
2.102 3 1 3.333
42.819 20 18 3.5
4.077 4 2 9
57.408 33 14 9.242
7.428 7 3 7
8.947 15 5 4