ПОЛТАВСЬКИЙ УНІВЕРСИТЕТ ЕКОНОМІКИ І ТОРГІВЛІ

Навчально-науковий інститут денної освіти

Форма навчання денна

Кафедра комп'ютерних наук та інформаційних технологій

Допускається до захисту

Завідувач кафедри

_____ Олена ОЛЬХОВСЬКА

«___» ____ 2023 p.

КВАЛІФІКАЦІЙНА РОБОТА

на тему

«РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ У ВИГЛЯДІ ВЕБ-ДОДАТКУ З ТЕМИ «ПОШУК ПАРАМЕТРІВ НЕЛІНІЙНОЇ РЕГРЕСІЇ» ДИСТАНЦІЙНОГО НАВЧАЛЬНОГО КУРСУ «ОБЧИСЛЮВАЛЬНІ МЕТОДИ»

зі спеціальності 122 Комп'ютерні науки

освітня програма «Комп'ютерні науки»

ступеня бакалавра

Виконавець роботи Терещенко Богдан Сергійович

______ «____»____ 2023 p.

Науковий керівник к.ф-м.н., доцент, Чілікіна Тетяна Василівна

______ «____»____ 2023 p.

Рецензент

2023

ЗАТВЕРДЖУЮ

Завідувач кафедри ____ Олена ОЛЬХОВСЬКА

(підпис)

«____» _____ 202___ p.

ЗАВДАННЯ І КАЛЕНДАРНИЙ ГРАФІК ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ

на тему «Розробка програмного забезпечення у вигляді веб додатку з теми «Пошук параметрів нелінійної регресії» дистанційного навчального курсу «Обчислювальні методи»

зі спеціальності 122 Комп'ютерні науки

освітня програма «Комп'ютерні науки»

ступеня бакалавр

Прізвище, ім'я, по батькові Терещенко Богдан Сергійович

Затверджена наказом ректора № <u>144</u>-Н від «<u>01</u>» <u>вересня</u> 2022 р.

Термін подання студентом роботи «____»____ 202__ р.

Вихідні дані до кваліфікаційної роботи: <u>публікації з теми, системи</u> дистанційного навчання.

Зміст пояснювальної записки (перелік питань, які потрібно розробити)

ВСТУП

РОЗДІЛ 1. ПОСТАНОВКА ЗАДАЧІ

1.1 Огляд програмних засобів для створення програмного забезпечення

РОЗДІЛ 2. ТЕОРЕТИЧНА ЧАСТИНА

- 2.1. Теоретичний матеріал за темою (нелінійна регресія та її характеристики).
- 2.2. Алгоритм роботи розробленого продукту.
- 2.3. Блок -схема.

РОЗДІЛ З. Практична частина

- 3.1. Обгрунтування елементів програмного забезпечення.
- 3.2. Опис створення елементів програмного продукту
- 3.3. Тестування створеного продукту.

ВИСНОВКИ

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

10		• 1 •	•••••••••••••••••••••••••••••••••••••••	<u>بر</u> .
Консультанти	розліців		ашинот	роботи
roneynbrainn	розділь	Rbusiipin	ацинон	pooorn

Розділ	Прізвище,	Підпі	ис, дата
	ініціали, посада консультанта	завдання видав	завдання прийняв
Постановка задачі	Чілікіна Т.В.		
Інформаційний огляд	Чілікіна Т.В.		
Теоретична частина	Чілікіна Т.В		
Практична частина	Чілікіна Т.В.		

Календарний	графік	виконання	кваліфіка	ційної	роботи
	1 1		1	1	1

Зміст роботи	Термін	Фактичне
	виконання	виконання
1. Вступ		
2. Вивчення методичних рекомендацій та стандартів		
та звіт керівнику		
3. Постановка задачі		
4. Інформаційний огляд джерел бібліотек та		
інтернету		
5. Теоретична частина		
6. Практична частина		
7. Закінчення оформлення		
8. Доповідь студента на кафедрі		
9. Доробка (за необхідністю), рецензування		

Дата видачі завдання «__» ____ 202___ р.

Здобувач вищої освіти _____ Терещенко Богдан Сергійович

(підпис)

Науковий керівник _____ к.ф-м.н., доц. Чілікіна Т.В.

(підпис)

(науковий ступінь, вчене звання, ініціали та прізвище)

Результати захисту кваліфікаційної роботи

Кваліфікаційна робота оцінен	а на		
	(балів, оцін	ка за національної	ю шкалою, оцінка за ЕСТЅ)
Протокол засідання ЕК №	від «	»	2023 p.
Секретар ЕК			
(підпис)		(ініціали та пр	різвище)

Затверджую

Зав. кафедрою _____

к.ф.-м.н. Олена ОЛЬХОВСЬКА

«____» _____ 202__ p.

Погоджено

Науковий керівник

к.ф-м.н., Тетяна ЧІЛІКИНА

«____» _____ 202__ p.

План

кваліфікаційної роботи на тему

«Розробка програмного забезпечення у вигляді веб додатку з теми «Пошук параметрів нелінійної регресії» дистанційного навчального курсу «Обчислювальні методи»

зі спеціальності 122 Комп'ютерні науки

освітня програма 122 «Комп'ютерні науки»

ступеня бакалавр

Прізвище, ім'я, по батькові Терещенко Богдан Сергійович

ВСТУП

1. РОЗДІЛ 1. ПОСТАНОВКА ЗАДАЧІ

1.1 Огляд програмних засобів для створення програмного забезпечення

2. РОЗДІЛ 2. ТЕОРЕТИЧНА ЧАСТИНА

- 2.1. Теоретичний матеріал за темою (нелінійна регресія та її характеристики).
- 2.2. Алгоритм роботи розробленого продукту.
- 2.3. Блок -схема.
- 3. Практична частина
 - 3.1. Обгрунтування елементів програмного забезпечення.
 - 3.2. Опис створення елементів програмного продукту
 - 3.3. Тестування створеного продукту.

ВИСНОВКИ

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

Здобувач вищої освіти _____ Богдан ТЕРЕЩЕНКО

«____» _____ 202__ p.

Реферат

Мета роботи - розробка програмного забезпечення у вигляді веб додатку для використання у дистанційному курсі.

Об'єкт розробки - Створення програмного забезпечення за темою «Пошук параметрів нелінійної регресії»

Методи дослідження - метод найменших квадратів, мова програмування Python, мови html i CSS

Було зроблено огляд теоретичної частини , що стосується саме «нелінійної регресії» та саме «МНК» . Після чого було прийнято рішення створення саме таблиці з підрахунками ,як основний контролюючий елемент ,який за допомогою Python і бібліотеки pandas вивантажується і перевіряється самим веб-додатком.

3MICT

ВСТУП

1. РОЗДІЛ 1. ПОСТАНОВКА ЗАДАЧІ

1.1 Огляд програмних засобів для створення програмного забезпечення.

2. РОЗДІЛ 2. ТЕОРЕТИЧНА ЧАСТИНА

- 2.1. Теоретичний матеріал за темою (нелінійна регресія та її характеристики).
- 2.2. Алгоритм роботи розробленого продукту.
- 2.3. Блок -схема.

3.РОЗДІЛ 3. Практична частина

- 3.1. Обгрунтування елементів програмного забезпечення.
- 3.2. Опис створення елементів програмного продукту.
- 3.3. Тестування створеного продукту.

Висновок.

Літературні джерела.

Вступ

Останнім часом, у зв'язку з швидкою комп'ютеризацією світової спільноти, зі створенням найскладнішої техніки, експлуатація якої пов'язана з ризиком для життя не лише однієї людини, але й людства в цілому, виникла ціла індустрія — тренажерні технології, які визначаються як складні комплекси, системи моделювання і симуляції, комп'ютерні програми та фізичні моделі, спеціальні методики, що створюються для того, щоб підготувати особистість до прийняття якісних і швидких рішень.

Метою цієї роботи є створення програмного забезпечення у вигляді веб додатку для «Пошуку нелінійної регресії» на основі метода найменших квадратів.

Для виконання цієї бакалаврської роботи потрібно виконати наступні завдання:

- Правильно поставити завдання для подальшого виконання;
- Детально прочитати інформацію з дистанційного курсу яка стосується «Нелінійної регресії»;
- Провести детальний інформаційний огляд даної теми;
- Створити алгоритм роботи веб-додатку;
- Створити блок-схему роботи додатка;
- Ретельно провести вибір мови та середовищ на яких буде розроблений додаток і обрати оптимальний варіант для забезбечення найкращого результату;
- Розробити сам додаток, його інтерфейс та інші складові які забезпечать гарний результат і забезпечать безперебійну роботу додатка;

 Провести тестування додатку у різних напрямках для виявлення його проблем(або іх відсутностей) і в разі потреби вирішити всі несправності.

Предметом бакалаврської роботи є тренажер для дисципліни « Обчислювальні методи» який має вигляд веб додатка сенс якого полягає у «Пошуку нелінійної регресії».

Використані методи в даній роботі:

- Для створення таблиці розрахунку було використано МНК який належить до теми «Нелінійна Регресія».

- А для реалізації програми у вигляді веб додатку було використано високорівневу мову Python. Для реалізації контролю над розрахунками була використана таблиця створена в Microsoft Excel. А для розробки дизайну інтерфейсу використовувалась мова html i CSS .

Ця робота є досить унікальною в тому ,що даних тренажерів з теми «Пошук нелінійної регресії» не існує ,тому ця робота має велику цінність і практичність бо вона полегшить процес навчання і дасть змогу студентам самореалізовуватися у даному напрямку.

Незвичним є і те який метод розробки був обраний. Створення основної таблиці яка просто вивантажує дані до програми є безперечною перевагою бо кожен зможе розібратися у роботі додатку і корегувати його під свої цілі ,навіть не маючи досвіду у роботі з даною темою, або навичків програмування на високому рівні. Тому ця робота буде безперечно цінною і рекомендована для використання студентів математичної та інших кафедр і напрямків навчання «Полтавського університету економіки та торгівлі».

Розділ 1. ПОСТАНОВКА ЗАДАЧІ

На сам перед ,головною метою цієї роботи є створення веб-додатку ,але незважаючи на це є не менш важливі завдання такі як:

- Вивчення і засвоєння теорії з курсу.
- Розробка самого програмного забезпечення.
- Розробка алгоритму роботи.
- Розробка блок-схеми.
- Обґрунтування вибору середовищ і мови.
- Тестування і виправлення помилок.

Але це далеко не всі вимоги до роботи. Є і вимоги до самого програмного забезпечення які потрібно виконати аби додаток вважався дійсно корисним і працездатним, а саме:

- Розробка самої таблиці для контролю обрахунків.
- Розробка програмного забезпечення використавши бібліотеки для полегшення роботи та збільшення функціоналу.
- Створення локального сервера за допомогою якого буде реалізований запуст додатку у веб-браузері і реалізація виводу повідомлень про виникнення помилок.
- Розробка простого але зрозумілого інтерфейсу користувача для додатка.
- Реалізація сторінки виводу результатів або виводу повідомлення про помилки .
- Тестування і виправлення помилок .

Розділ 1.1. Огляд програмних засобів для створення програмного забезпечення

Для створення свого веб-додатку я обрав мову Python з деякими бібліотеками які полегшують роботу. Excel був використаний лля створення таблиці, котрі слугують для зміни даних які будуть обчислюватися додатком і загалом слугує умовним ЦУ для цього додатка через який можна маніпулювати підрахунками і варіативно змінювати параметри для знаходження нелінійної регресії. Також можлива зміна кількості параметрів для підрахунку і це далеко не всі можливості і варіанти розвитку котрі можна реалізувати. А для того аби зробити додаток естетичним було обрано CSS для розробки простого але інтуітивно зрозумілого інтерфейсу.

Python є мультипарадигменною мовою програмування, що підтримує імперативне, процедурне, структурне, об'єктно-орієнтоване програмування, метапрограмування функціональне програмування Завдання та узагальненого програмування вирішуються рахунок динамічної типізації. Аспектно-орієнтоване програмування частково підтримується через декоратори, повноцінна підтримка забезпечується додатковимифреймворками. Такі методики як контрактне та логічне програмування можна реалізувати за допомогою бібліотек чи розширень. Основні архітектурні риси - динамічна типізація, автоматичне управління пам'яттю, повна інтроспекція, механізм обробки винятків, підтримка багатопоточних обчислень з глобальним блокуванням інтерпретатора, високорівневі структури даних. Підтримується розбиття програм на модулі, які, своєю чергою, можуть об'єднуватися в пакети.



Рис. 1.1.1

Microsoft Excel 2010 – для розробки таблиці з якої будуть вивантажуватися дані для обчислення , А також буде виконуватися основний процес контролю, заміни та інших процесів пов'язаних з обрахунками у вебдодатку.

Excel — програмований табличний калькулятор. Всі розрахунки в Excel виконують формули. Excel вважає формулою все, що починається зі знаку "=". Якщо в комірці написати просто «1 +1», то Excel не буде обчислювати цей вираз. Однак, якщо написати «=1 +1» і натиснути клавішу Enter, в комірці з'явиться результат обчислення виразу — число 2. Після натискання клавіші Enter формула не пропадає, її можна побачити в панелі інструментів «Рядок формул».

У формулі можна використовувати різні типи операторів (арифметичні і т. ін.), текст, посилання на комірку або діапазон комірок, круглі дужки, іменовані діапазони. Природно, в формулах дотримується пріоритет виконання операцій (множення виконується раніше додавання і т. ін.). Для зміни порядку виконання операцій використовуються круглі дужки.

Для того, щоб вставити у формулу адресу клітинки (посилання на клітинку), не обов'язково писати його вручну. Простіше поставити знак "=", потім лівою кнопкою клацнути на потрібній клітинці або виділити потрібний діапазон клітинок. При цьому Ехсеl підставить у формулу посилання автоматично. Якщо у формулі використовується декілька посилань, то

14

кожній з них Excel дає свій колір. Це дуже зручно. Приклад: напишіть в будьякій клітинці формулу «= A1 + D1», натисніть Enter, потім двічі клацнути по клітинці. Тоді ви побачите формулу з різнокольоровими посиланнями, а навколо клітинок A1 і D1 будуть прямокутники відповідних кольорів. Набагато простіше знайти. куди вказує посилання. за кольором прямокутника, ніж переглядати букви стовпців і номери рядків. Наведіть курсор миші на один з різнокольорових прямокутників і перетягніть лівою кнопкою за рамку в інше місце. Ви побачите, що при цьому міняються відповідно і адреси клітинок у формулі — часто це найскоріший спосіб підправити адресу у формулі, особливо після копіювання маркером автозаповнення. Коли треба цьому запобігти, достатньо поставити знак долара (\$) перед рядком або стовпчиком.

Наприклад, якщо в клітинці ВЗ є формула, виражена через клітинку А5, то в С6 ця клітинка стане В8, \$А5 стане А8, А\$5 стане В5, а \$А\$5 не зміниться.



Рис.1.1.2

Середовище CSS3 та html- для розробки дизайну та інтерфейсу веб-додатку.

CSS використовується творцями веб-сторінок для завдання кольорів , шрифтів , стилів, розташування окремих блоків та інших аспектів представлення зовнішнього вигляду цих веб-сторінок. Основною метою розробки CSS є огородження та відокремлення опису логічної структури вебсторінки (яке виробляється за допомогою HTML або інших мов розмітки) від опису зовнішнього вигляду цієї веб-сторінки (яке тепер виробляється за допомогою формальної мови CSS). Такий поділ може збільшити доступність документа, надати більшу гнучкість та можливість управління його поданням, а також зменшити складність та повторюваність у структурному вмісті.

Крім того, CSS дозволяє представляти один і той же документ у різних стилях або методах виводу, таких як екранне подання, друковане подання, читання голосом (спеціальним голосовим браузером або програмою читання з екрану) або при виведенні пристроями, що використовують шрифт Брайля.



Рис 1.1.3

HTML - тегова мова розмітки документів . Будь-який документ мовою HTML є набором елементів , причому початок і кінець кожного елемента позначається спеціальними позначками - тегами . Елементи можуть бути порожніми , тобто не містять жодного тексту та інших даних. У цьому випадку зазвичай не вказується тег, що закриває (наприклад, тег перенесення рядка
 - одиночний і закривати його не потрібно) . Крім того, елементи можуть мати атрибути , що визначають якісь властивості (наприклад, атрибут href="y посилання). Атрибути вказуються у тезі, що відкриває. Ось приклади фрагментів HTML-документу:

За допомогою HTML різні конструкції, зображення та інші об'єкти, такі як інтерактивна веб-форма, можуть бути вбудовані в сторінку, що відображається. HTML пропонує засоби для створення заголовків, абзаців, списків, посилань, цитат та інших елементів. Елементи HTML виділяються тегами, записаними з використанням кутових дужок. Такі теги, як i <input />, безпосередньо вводять контент на сторінку. Інші теги, такі як , оточують і оформляють текст всередині себе і можуть включати інші теги як поделементи. Браузери не відображають HTML-теги, але використовують їх для інтерпретації вмісту сторінки.

Текстові документи, що містять розмітку на мові HTML (такі документи традиційно мають розширення .html або .htm), обробляються спеціальними додатками , які відображають документ у форматованому вигляді. Такі програми, що називаються «браузерами» або «браузерами», зазвичай надають користувачеві зручний інтерфейс для запиту веб-сторінок, їх перегляду (і виведення на інші зовнішні пристрої) і, при необхідності, відправки введених користувачем даних на сервер. Найбільш популярними на сьогоднішній день браузерами є Google Chrome, Opera та інші.



Рис. 1.1.

Розділ 2 Теоретична частина

Розділ 2.1 Теоретичний матеріал за темою

«Нелінійна регресія»

Нелінійна регресія - це вид регресійного аналізу , в якому експериментальні дані моделюються функцією, що є нелінійною комбінацією параметрів моделі і залежить від однієї і більш незалежних змінних. Дані апроксимуються методом послідовних наближень.

Загальні положення

Дані складаються з вільних від помилок незалежних змінних х і пов'язаних залежних змінних (відгуків) у , що спостерігаються . Кожна змінна у моделюється як випадкова величина із середнім значенням , що задається нелінійною функцією. Методична похибка може бути, але її обробка виходить за межі регресійного аналізу. Якщо незалежні змінні не вільні від помилок, модель стає моделлю з помилками в змінних і також виходить за рамки розгляду.

Види регресій

Поліномальне рівняння регресії

$y = a + bx + cx^2$

Гіперболічне рівняння регресії

$$y_t = a + b\frac{1}{x_t} + e_t$$

Квадратичне рівняння регресії

$y = a + b\sqrt{x}$

Данні методи нелінійні щодо включених в аналіз змінних, але лінійні за оцінюваними параметрами. Також є методи які нелінійні за оцінюваними параметрами, такі як:

Показове рівняння регресії

$y_{t=}ab^{x_t}\varepsilon_t$

Експонентне рівняння регресії , Ступінне рівняння регресії , Напівлогарифмічне рівняння регресії котрі нелінійні за всіма оцінюваними параметрами.

Метод найменших квадратів (МНК)

Метод найменших квадратів (МНК) - математичний метод, що застосовується для вирішення різних завдань, що базується на мінімізації суми квадратів відхилень деяких функцій від експериментальних вхідних даних. Він може використовуватися для «вирішення» перевизначених систем рівнянь (коли кількість рівнянь перевищує кількість невідомих), для пошуку рішення у разі звичайних (не перевизначених) нелінійних систем рівнянь, для апроксимації точкових значень певної функції. МНК є одним із базових методів регресійного аналізу для оцінки невідомих параметрів регресійних моделей за вибірковими даними.

Регресійний аналіз — розділ математичної статистики, присвячений методам аналізу залежності однієї величини від іншої. На відміну від кореляційного аналізу не з'ясовує істотний зв'язок, а займається пошуком моделі цього зв'язку, вираженої у функції регресії.

Ці два поняття є дуже тісно пов'язані між собою ,бо саме за допомогою цього метода (МНК) можна точно і безпомилково знайти залежність параметрів одне від одного, незважаючи на невідомі параметри і похобки. Наприклад ми маємо п значень деякої змінної y і відповілної змінної х. Завдання в тому , щоб взаємозв'язок між у та х апроксимувати деякою функцією f(x,b) де b це невідомий параметр. Тобто задача в тому ,щоб знайти найкращі параметри значення параметру b при яких значення f(x,b) будуть максимально наближені до фактичних значень y. Тобто виглядати це буде так: $f(x_t, b) = y_t, t = 1, ..., n$

У регресійному аналізі та зокрема в економетриці використовуються ймовірнісні моделі залежності між змінними

$$y_t = f(x_t b) + \varepsilon_t$$

де ε_t це ті самі випадкові помилки моделі.

Також можна відмітити те, що сам принцип нелінійної регресії можна застосувати не тільки для задач з математики, економіки і вирішення фінансових завдань, а і д ля таких речей як ферментативна кінетика.

Таким чином можна записати модель Михаєлиса – Ментена, для ферментативної кінетики, яка виглядає так :

$$v = \frac{V_{max}[S]}{K_m + [S]}$$

I перетворити її на модель нелінійної регресії, яка буде виглядати так:

$$f(x,\beta) = \frac{\beta_1 x}{\beta_2 + x}$$

В цьому рівнянні β_1 відповідає параметру V_{max} , параметр β_2 таким чином відповідає K_m , а [S] заміняється параметром х.

Розділ 2.2 АЛГОРИТМ РОБОТИ РОЗРОБЛЕНОГО ПРОДУКТУ

Крок 1. Перед запуском програми потрібно здійснити налаштування в таблиці ,але якщо це не зробити то відповідь буде такою як і розрахунки які заздалегідь зроблені в таблиці.

Крок 2. Після запуску додатка користувач побачить три поля для вводу даних.

Крок З.Користувач повинен ввести дані у пусті поля

Крок 4. Користувачу потрібно натиснути на зелену кнопку « Показати результат» .

Крок 5.Після натискання на кнопку користувач перейде до сторінки з перевіркою відповідей. У випадку правильності відповідей користувач побачить повідомлення біля кожного рядка з даних. У випадку коли відповідь не правильна також буде показано повідомлення.

Крок 6. У випадку потреби зміни відповіді потрібно змінити числові дані у таблиці і повторити попередні кроки.

Розділ 2.3 БЛОК-СХЕМА



Розділ З ПРАКТИЧНА ЧАСТИНА Розділ З.1. ОБҐРУНТУВАННЯ ВИБОРУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Для написання коду була обрана мова Python, тому ,що стояло завдання створити веб додаток, а для цього потрібно запустити локальний сервер. Це можливо зробити багатьма способами , але за допомогою Python і мікрофреймворка Flask запуск сервера не є складністю і не викликає жодних проблем. І сама велика перевага цього методу це легке підключення і застосування з можливістю подальшого розвитку з локального сервера до більш глобального проекту. Друга причина чому була обрана мова Python це бібліотека pandas за допомогою якої можливо витягнути дані з таблиці для обрахунків і використати їх для перевірки і повернення результатів у додатку. Також ця бібліотека слугує загалом для роботи із файлами і дуже полегшує життя і роботу в проектах ,де вам потрібно щось завантажити або вивантажити використовуючи сторонні файли.

Для розробки простого але приємного ,для користувача , інтерфейсу було використано html та CSS через доступність і простото у використанні. Найбільшим плюсом використання цього набору в тому, що розробити інтерфейс ,особливо статичний, найпростіше саме тут і одразу можна зберегти дизайн у декількох форматах для простоти накладення його на ваш проект. Тут можна не боятися розпочати серйозний проект ,навіть не маючи великої кількості досвіду . Тому саме тут був розроблений статичний дизайн який був розділений на дві частини (основну та частину з виводом даних) для простоти і зручності накладення на додаток і для можливості зміни однієї частини у випадку потреби або помилки при кодуванні.

Для створення основного елементу контролю всіх підрахунків було створено таблицю в Microsoft Excel. Цей елемент відповідає за усі процеси пов'язані с розрахунками та слугує для детального налаштування додатку під кожного користувача бо саме тут можна змінити усі елементи які пов'язані з кінцевим результатом виводу в додатку. На цьому можливості налаштування не закінчуються бо варіативність цього тренажера обмежується лише завданням і бажанням користувача. Я обрав такий варіант через відносну простоту створення, варіативність обчислень без великої кількості змін і через можливість розрахувати і транспонувати формули з різних областей

Розділ 3.2. Опис створення елементів програмного забезпечення

Для початку створимо таблицю яка виглядатиме так:

1	A	В	С	D	E	F	G	Н	1	J	K	L	Μ	N	0	Р	Q	R	S	Т	
1	Метод на	айменших н	квадратів																		
2		Розрахун	кова табли	ця					лінійна		квадрати	ина									
3	1	Xi	Yi	Xi ²	Xi ³	Xi ⁴	XiYi	Xi ² Yi	лін	кв.відхил.	квадр.	кв.відхил.									
4	1	0,50	1,00	0,25	0,13	0,06	0,50	0,25			1,29										
5	2	1,00	3,00	1,00	1,00	1,00	3,00	3,00			2,26										
6	3	1,50	2,00	2,25	3,38	5,06	3,00	4,50			2,51										
7	4	2,00	2,00	4,00	8,00	16,00	4,00	8,00			2,06										
8	5	2,50	1,00	6,25	15,63	39,06	2,50	6,25			0,89										
9		7,50	9,00	13,75	28,13	61,19	13,00	22,00	0,00	0,00	9,00	0,00									
10																					
11													Квадрат	ична апрок	симація у=а	x2+bx+c					_
12														Система р	оівнянь MHI	{					-
13														a	b c		пр.част				
14														61,19	28,13	13,75	22,00				
15														28,13	13,75	7,50	13,00				
16														13,75	7,50	5,00	9,00				
17																					
18													Рішенн	системи	a=	-1,43					
19															b=	4,09					
20															C=	-0,40					
21																					

Рис. 3.2.1

Заповнюємо колону яка знаходиться під X_ii Y_i даними і в останньому рядку стовпчика X_i вводимо формулу яка виглядає ось так:



Рис 3.2.2

(=СУММ(В4:В8)) – виконує підрахунки усіх введених параметрів починаючи з В4 по В8 включно. У моєму випадку вийшло число 7,50, але якщо змінити данні то відповідь буде різна залежно від того ,що ви там введете. Таку ж процедуру робимо з Y_i а формула виглядає так:

fx =CYMM(C4:C8)

Рис. 3.2.3

(=СУММ(С4:С8)) – тут я отримав відповідь 9.

Таким самим чином заповнюємо і інші стовпці $x_i^1, x_i^2, x_i^3, x_i^4$. Але в стовпчику зі степенями потрібно записати як (=Bx^n) де n це степінь а x це номер рядку(починаючи з 4 закінчуючи 8)

Від першого:

=B4^2 fx

Рис. 3.2.4

До останнього рядка стовпчика :

*f*_{*} =B8^2

Рис. 3.2.5

В кінці та ж формула

fx =CYMM(D4:D8)

Рис. 3.2.6

Таку процедуру робимо з усіма стовпцями де є степінь.

Далі у нас йде стовпець $X_i Y_i$, який заповнюється таким чином:



Рис. 3.2.7

Тобто загальна формула виглядатиме так : (=Bn* Cn) (від 4 до 8) де кожному рядку В належить рядок С с таким же самим номером.

Далі по черзі йде стовпчик X_I²Y_iякий заповнюється так:

Рис. 3.2.8

(=Bx^2*Cx) де х це номер рядка (від 4 до 8) і заповнюється так само як і попередній. Стовпець С повинен відповідати стовпцю В по номерам рядків.

Вигляд формули останнього рядка у стовпчику:

fx =CYMM(H4:H8)

Рис. 3.2.9

Далі переходи до заповнення останнього стовпчика

А	В	С	D	E	F	G	Н	1	J	K	L
Иетод на	айменших н	квадратів									
	Розрахуни	кова табли	ця					лінійна		квадрати	чна
I.	Xi	Yi	Xi ²	Xi ³	Xi ⁴	XiYi	Xi ² Yi	лін	кв.відхил.	квадр.	кв.відхил
1	0,50	1,00	0,25	0,13	0,06	0,50	0,25			1,29	
2	1,00	3,00	1,00	1,00	1,00	3,00	3,00	/		2,26	
3	1,50	2,00	2,25	3,38	5,06	3,00	4,50			2,51	
4	2,00	2,00	4,00	8,00	16,00	4,00	8,00			2,06	
5	2,50	1,00	6,25	15,63	39,06	2,50	6,25			0,89	
	7,50	9,00	13,75	28,13	61,19	13,00	22,00	0,00	0,00	9,00	0,00

Рис. 3.2.10

Для його заповнення використаємо формулу яка виглядає таким чином:

fx =\$P\$18*B4^2+\$P\$19*B4+\$P\$20

Рис. 3.2.11

(=\$P\$18*Bx^2+\$P\$19*Bx+\$P\$20) – де х це номер рядка починаючи з 4 і закінчуючи 8.

Ця формула виконує останні розрахунки і виводить їх у спеціальне поле в якому буде знаходитись саме рішення системи.

Рішення системи	a=	-1,43
	b=	4,09
	c=	-0,40

Рис. 3.2.12

Таким чином ми розібралися як будуть виконуватися розрахунки і яку саме відповідь буде перевіряти наш додаток, а також дізналися яким чином можна змінити дані які будуть обраховуватися на ті ,які потрібні саме вам. Три кінцевих значення (-1,43, 4,09, -0,40) будуть перевірені в самому додатку і якщо це будуть не правильні дані то система вам про це повідомить.

Тепер переходимо до розробки самого додатку



Рис. 3.2.13

Тут застосований мікрофреймворк Flask ,для створення і можливості запуску свого локального сервера .



Рис. 3.2.13

В цьому блоці розписаний запуск і завантаження сторінки веб-додатку де буде виконуватися перевірка даних. У випадку коли трапилась помилка або не корректно був запущений локальний сервер або через будь яку іншу помилку,через яку не вдалося завантажити сторінку ,ви отримаєте таке повідомлення «Виникла помилка при завантаженні сторінки» і це буде сигналом, що виникла проблема ,скоріш за все с мікрофреймворком і потрібно її вирішити.



Рис. 3.2.14

Функція яку ми оголошуємо як приймач запитів ,пишучі @app.route. Далі пишемо роут у нашому випадку це /answer та метод запиту post. В тілі функції. Далі за допомогою конструкції try exept витягуємо дані і відображуємо їх у новій сторінці.

Рис. 3.2.15

Тут ,за допомогою бібліотеки Pandas, загружаємо дані з таблиці в якій ми зробили розрахунки.





А вже тут, ми конкретно витягуємо дані a,b,c і форматуємо їх до 2-х значень після знаку для полегшення роботи з ними і більш компактного виводу, але якщо вам потрібна більша точність ви можете змінити значення з 2 на 4 чи 5, або будь яке інше значення ,котре вам потрібно.



Рис. 3.2.17

Тут проходить процес заповнення лани і завантажуються в об'єкт, після чого вони відправляються на кінцеву сторінку з результатами яку буде перевіряти наш додаток і звірювати з даними котрі ми заповнювали а далі вивантажили.





Тут ми можемо побачити як проходить процес загрузки сторінки з результатами і як система загружає файл з дизайном кінцевої сторінки за допомогою тієї ж самої бібліотеки .У випадку помилки з завантаженням сторінки через проблему з сервером або з проблемою перевірки роута повертається повідомлення про помилку ,як і у випадку з помилкою роботи мікрофреймворка.

# Запуск web додатка	
app.run()	

Рис. 3.2.19

Ну і останнє це сам запуск веб додатку.

Рис. 3.2.20

Так виглядає сам дизайн інтерфейса користувача. Він знаходиться у папці з додатком і має ім'я index і розширення html. Це означає ,що кожен може відкрити цей файл за допомогою звичайного браузера для перевірки його зовнішнього вигляду і потім ,у разі потреби, щось змінити ,або взагалі змінити весь файл на свій дизайн ,але потрібно пам'ятати ,що потрібно його зберегти під таким самим ім'ям, або змінити ім'я в коді, ось тут:



Рис. 3.2.21

A саме рядок «return render_template ('index.html') і поставити тут ім'я свого файлу.

Наприклад ««return render_template ('MyDesign.html') і обов'язково слід пам'ятати те що ім'я файлу повинно бути написано латиницею без використання українських літер, бо це скоріш за все приведе к не коректному відображенню або взагалі поверне вам помилку.

Якщо запустити саму сторінку без програми і натиснути на кнопку «Показати результат» то ми отримаємо помилку яка виглядатиме так:



Рис. 3.2.22

Ця помилка викликана тим, що це лише оболонка сайту і вона не виконує ніяких дій окрім самого зовнішнього виду.

Так само можемо подивитися який вигляд має сторінка з кінцевим

результатом

Результат

{% if user_value.a == user_value.table_a %}
(A) {{user_value.a}} Правильна відповідь {% else %}
(A) {{user_value.a}} Неправильна відповідь {% endif %} {% if user_value.b == user_value.table_b %}
(B) {{user_value.b}} Правильна відповідь {% else %}
(B) {{user_value.b}} Неправильна відповідь {% endif %} {% if user_value.c == user_value.table_c %}
(C) {{user_value.c}} Правильна відповідь {% else %}
(C) {{user_value.c}} Неправильна відповідь {% else %}
(C) {{user_value.c}} Неправильна відповідь {% endif %}

Рис. 3.2.23

Тут ми можемо побачити усі можливі результати виводу. І як раз тому, що ми не завантажили локальний сервер і не запустили саму програму "ми не змогли перевірити результати у попередньому випадку з головною сторінкою. Тому потрібно завантажити додаток правильно аби все працювало коректно і без помилок, а як саме це зробити невдовзі покажу і поясню.

Зараз подивимося на то як саме дизайн виглядаю з точки зору кодування.

Відкриваємо файл «styles.css» ,який відповідає файлу index.html.

```
body {
    font-family: Arial, sans-serif;
    background-color: #f1f1f1;
    margin: 0;
    padding: 0;
}
div {
    max-width: 800px;
    margin: 0 auto;
    padding: 20px;
    background-color: #fff;
    box-shadow: 0 0 5px rgba(0, 0, 0, 0.1);
}
h1 {
    color: #333;
    text-align: center;
}
input[type="text"] {
    width: 780px;
    padding: 10px;
    font-size: 16px;
    border: 1px solid #ccc;
    border-radius: 4px;
3
```

Рис. 3.2.24

На цій частині коду ми бачимо як влаштований сам дизайн. Ширина тексту, розміри самих полів для вводу,який саме шрифт використовується для відображення тексту. Усе тут можна змінити під свої потреби починаючи від кольорів сайту до написів на полях для вводу, їх розміри та все інше.

```
porder-radius: 4px;
}
button {
    padding: 10px 20px;
    font-size: 16px;
    background-color: #4CAF50;
    color: #fff;
    border: none;
    border-radius: 4px;
    cursor: pointer;
}
button:hover {
    background-color: #45a049;
}
button:active {
    background-color: #3e8e41;
}
button:focus {
    outline: none;
}
```

Рис. 3.2.25

А в цій частині коду ми можемо побачити як буде реагувати кнопка при натиску на неї, проходженні курсора через неї, кордони самої кнопки і реагування цих кордонів і багато іншого. Навіть колір кнопки можна змінити лише одним рядком. Також при натисканні на кнопку вона трохи змінює свій колір на більш темний, що додає реалізму і відчуття, що кнопка дійсно була натиснута. Тому якщо вам не подобається дизайн ви можете його легко змінити під себе.

```
body {
  display: flex;
  justify-content: center
 align-items: center;
 height: 100vh;
 margin: 0;
  font-family: Arial, sar
}
.container {
  text-align: center;
}
h1 {
 font-size: 24px;
 margin-bottom: 20px;
}
p {
  font-size: 18px;
  margin-bottom: 10px;
}
```

Рис. 3.2.26

А так виглядає сторінка з результатами. Ми можемо побачити ,що тут не так багато налаштувань, і всі вони виглядають приблизно як і те,що ми розглянули до цього. Також можна змінити параметри тексту і шрифту.

Розділ 3.3. Тестування створеного продукту

Для початку потрібно вивантажити папку з архіву.



Рис.3.3.1

Потім зайти в папку з додатком

1мя	Дата изменения	Тип	Размер
static	30.05.2023 17:37	Папка с файлами	
tables	30.05.2023 17:46	Папка с файлами	
templates	30.05.2023 17:45	Папка с файлами	
gitignore	30.05.2023 17:49	Файл "GITIGNORE"	1 КБ
👌 main.py	30.05.2023 17:44	Python File	2 КБ
requirements.txt	30.05.2023 17:48	Текстовый докум	1 KĐ

Рис.3.3.2

Далі нам потрібно запустити cmd за допомогою команди win+r викликати вікно для швидкого запуску ввести команду і натиснути кнопку «ОК»



Рис.3.3.3

Після цього ви побачите таке вікно



Рис.3.3.4 Одразу перевіримо версію Python, нам потрібна версія не менше 3



Бачимо ,що у нас все добре і переходимо до наступного кроку ,якщо ні то потрібно зробити наступне:

1.Зайти на офіційний сайт Python.org



Рис.3.3.6

2.Натискаємо кнопку «Downloads» або « Заватнажити»



Рис.3.3.7



Рис.3.3.8

Після чого проводимо процес встановлення даного програмного забезпечення і після чого повертаємося до попередніх кроків.

Коли ми впевнені ,що у нас встановлена версія 3+ переходимо до папки з програмою і копіюємо шлях до папки.

мя	Дата изменения	Тип	Размер
📑 static	30.05.2023 17:37	Папка с файлами	
tables	04.06.2023 13:58	Папка с файлами	
templates	30.05.2023 17:45	Папка с файлами	
📑 .gitignore	30.05.2023 17:49	Файл "GITIGNORE"	1 КБ
🗟 main.py	30.05.2023 17:44	Python File	2 КБ
requirements.txt	30.05.2023 17:48	Текстовый докум	1 КБ

Рис.3.3.9

I потім переходимо до консолі і вводимо команду «cd C:\Users\NAME\ \UniversityTASK» і вставляємо шлях до папки і натискаємо ентер



Рис.3.3.10

Далі нам потрібно завантажити venv де знаходяться скріпти для активації додатку ,а це можна зробити командою «python –m venv myvenv(за допомогою якої буде створена папка в якій знаходяться основні файли для

активації програми.



Рис.3.3.11

Ось ця сама папка

myyeny	04.06.2023 16:34	Папка с файлами	
static 🖌	30.05.2023 17:37	Папка с файлами	
ables	04.06.2023 13:58	Папка с файлами	
templates	30.05.2023 17:45	Папка с файлами	
📑 .gitignore	30.05.2023 17:49	Файл "GITIGNORE"	1 КБ
🗟 main.py	30.05.2023 17:44	Python File	2 КБ
i requirements.txt	30.05.2023 17:48	Текстовый докум	1 KG



Далі вводимо команду « .\myvenv\Scripts\activate»



Після цього ми активували наш скріпт ,тепер нам потрібно встановити потрібні пакети Python за допомогою команди «pip install -r .\requirements.txt» яка завантажує потрібні для роботи програми .



Рис.3.3.14



Рис.3.3.15

Доки у нас завантажуються потрібні пакети і бібліотеки, можемо подивитися що саме ми завантажили відкривши той самий файл «requirements.txt»

```
blinker==1.6.2
certifi==2023.5.7
charset-normalizer==3.1.0
click==8.1.3
colorama==0.4.6
et-xmlfile==1.1.0
Flask==2.3.2
idna==3.4
itsdangerous==2.1.2
Jinja2==3.1.2
MarkupSafe==2.1.2
numpy==1.24.3
openpyx1==3.1.2
pandas==2.0.2
python-dateutil==2.8.2
pytz==2023.3
six==1.16.0
tzdata==2023.3
urllib3==2.0.2
Werkzeug==2.3.4
```

Рис.3.3.16

Загрузка бібліотек і пакетів триватиме доволі довго якщо ви їх ніколи не встановлювали. Тоді можемо подивитися на інші файли папки доки йде завантаження

myvenv	04.06.2023 16:34	Папка с файлами	
static	30.05.2023 17:37	Папка с файлами	
tables	04.06.2023 13:58	Папка с файлами	
templates	30.05.2023 17:45	Папка с файлами	
🗋 .gitignore 🔪	30.05.2023 17:49	Файл "GITIGNORE"	1 K6
amain.py	30.05.2023 17:44	Python File	2 КБ
requirements.txt	30.05.2023 17:48	Текстовый докум	1 KE

Рис.3.3.17

Имя	Дата изменения	Тип	Размер
🧿 index.html	30.05.2023 17:41	Chrome HTML Do	1 КБ
🧿 result.html	30.05.2023 17:45	Chrome HTML Do	2 КБ

Рис.3.3.18

У цій папці знаходиться наш дизайн про який ми говорили раніше.

- myvenv	04.06.2023 16:34	Папка с файлами	
static	30.05.2023 17:37	Папка с файлами	
tables	04.06.2023 16:11	Папка с файлами	
tern lates	30.05.2023 17:45	Папка с файлами	
gitignore	30.05.2023 17:49	Файл "GITIGNORE"	1 КБ
🗟 main.py 🔪	30.05.2023 17:44	Python File	2 КБ
requirements.txt	30.05.2023 17:48	Текстовый докум	1 КБ



Подивимося ,що у цій папці



Рис.3.3.20

Саме тут у нас та сама таблиця через яку проводяться усі розрахунки і дані з якої вивантажуються додатком для перевірки відповіді.

Нарешті все завантажилося і ми бачимо повідомлення про нову версію і команду для оновлення, але зараз це нам не потрібно, але якщо хочеться то можна оновитися.



Рис.3.3.21

I саме останнє що нам залишилось зробити – це запустити наш додаток

командою «.\main.py»

```
(myvenv) C:\Users\Jenya\Desktop\UniversityTASK>.\main.py
 * Serving Flask app 'main'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
```

Рис.3.3.22

Ми отримали айпі адресу ,котру нам потрібно скопіювати і вставити в

адресну строку браузера і натиснути ентер.





«Цю процедуру бажано виконати лише при першому запуску ,а в подальшому можна запускати програму через файл «Start,bat».»

Відкривається наша сторінка

а	
b	
C	
Перевірити результат	

Рис.3.3.24

Вводимо наші данні для перевірки

-1.43			
4.09			
-0.40			
Перевірити результ	ат		

Рис.3.3.25

Натискаємо ентер і дивимося на результати

Результат

(А) -1.43 Правильна відповідь

(В) 4.09 Правильна відповідь

(С) -0.40 Правильна відповідь

Рис.3.3.26

Тепер давайте введемо інші данні для перевірки додатку.

Результат

1 100 I I I I

- (А) -1.11 Неправильна відповідь
- (В) 4.09 Правильна відповідь
- (С) -0.40 Правильна відповідь

Рис.3.3.27

Як бачите при зміні параметра нам показало що відповідь неправільна.

Давайте змінимо всі параметри і подивимося результат.

Введіть значення (a, b, c) для перевірки

Округліть значення до 2 знаків після крапки. Приклад 1.23

-1.11

5.09

-1.40

Перевірити результат



Результат

- (А) -1.11 Неправильна відповідь
- (В) 5.09 Неправильна відповідь
- (С) -1.40 Неправильна відповідь

Рис.3.3.29

Як бачимо все чудово працює. Тому можна тренуватися в розрахунках.

Висновок

У результаті виконання бакалаврської дипломної роботи були виконані усі поставлені на початку завдання, а саме:

- Розробка самого програмного забезпечення.
- Розробка алгоритму роботи.
- Розробка блок-схеми.
- Обґрунтування вибору середовищ і мови.
- Розробка самої таблиці для контролю обрахунків в Microsoft Excel.
- Розробка програмного забезпечення використавши бібліотеки pandas,
 Jinja2, colorama, idna, click та інші, для полегшення роботи та збільшення функціоналу.
- Створення локального сервера за допомогою Python і бібліотеку Flask
 Через яку було реалізований запуск додатку у веб-браузері і реалізація виводу повідомлень про виникнення помилок.
- Розробка простого але зрозумілого інтерфейсу користувача для додатка використавши CSS та мову html.
- Реалізація сторінки виводу результатів або виводу повідомлення про помилки .
- Тестування і виправлення помилок.

СПИСОК ЛІТЕРАТУРИ

- Що таке мова Python? [Електронний ресурс]. –Режим доступу до ресурсу: URL:<u>https://freehost.com.ua/ukr/faq/wiki/chto-takoe-jazik-programmirovanijapython/</u> – Назва з екрану.
- 2. Вивчення нелінійної регресії [Електронний ресурс]. Режим доступу до ресурсу: URL: https://ua.nesrakonk.ru/nonlinear-regression/ Назва з екрану.
- What is nonlinear regression? [Електронний ресурс]. Режим доступу до ресурсу: URL: https://ivestopedia.com/tems/n/nonlinear-regession.asp Назва з екрану.
- 4. Нелінійна регресія. Основні поняття і застосування [Електронний книга].
 –Режим доступу до ресурсу: URL: https://evnuir.vnu.edu.ua/bitstream/123456789/17889/1/Nonlinear%20regressio n_Begun_Muran.pdf – Назва з екрану.
- 5. What is CSS? [Електронний ресурс]. Режим доступу до ресурсу: URL: https://developer.mozilla.org/en-

US/docs/Learn/Getting_started_with_the_web/CSS_basics- Назва з екрану.

- 6. Створення веб-додатку на Python 3 [Електронний ресурс]. Режим доступу до ресурсу: URL: https://www.digitalocean.com/community/tutorials/how-to-make-a-web-application-using-flask-in-python-3-ru Назва з екрану.
- 7. Введення в бібліотеку pandas.Python: [Електронний ресурс]. Режим доступу до ресурсу: URL https://khashtamov.com/ru/pandas-introduction/– Назва з екрану.
- Метод найменших квадратів [Електронний ресурс]. Режим доступу до pecypcy: URL: https://ua.nesrakonk.ru/least-squares-method/– Назва з екрану.
- 9. Python [Електронний ресурс]. Режим доступу до ресурсу: URL: Python.org Назва з екрану.

10. CSS підручник [Електронний ресурс]. – Режим доступу до ресурсу: URL: https://w3schoolsua.github.io/css/index.html#gsc.tab=0