**Київський національний університет імені Тараса Шевченка**

**Інститут філології**

**Кафедра української мови та прикладної лінгвістики**

Лабораторні роботи

з квантитативної лінгвістики

Виконала

студентка 3-го курсу

ОПП "Прикладна (комп'ютерна)

лінгвістика та англійська мова"

Черниш Леся Володимирівна

Київ - 2021

ЗМІСТ

[**Використані джерела**](#_heading=h.gjdgxs)

[**РОЗДІЛ 1. Створення електронного частотного словника**..........................3](#_heading=h.30j0zll)

**1.** [**Формування текстової вибірки**](#_heading=h.1fob9te)………………………………………....4

[**2. Укладання ЧС**](#_heading=h.3znysh7)……………………………………………………………..4

[**3. Частотні словники**](#_heading=h.2et92p0)………………………………………………………14

[**4. Характеристика вибірок**](#_heading=h.tyjcwt)……………………………………………...17

[**5. Полігони частот**](#_heading=h.3dy6vkm)…………………………………………………………18

**1.5.1.**[**Іменник**](#_heading=h.1t3h5sf)……………………………………………………………..18

**1.5.2.** [**Прикметник**](#_heading=h.4d34og8)……………………………………………………….20

**1.5.3.** [**Дієслово**](#_heading=h.2s8eyo1)……………………………………………………………22

**1.5.4.** [**Займенник**](#_heading=h.17dp8vu)………………………………………………………….25

**1.5.5.** [**Сполучник**](#_heading=h.3rdcrjn)…………………………………………………………28

**1.5.6.** [**Числівник**](#_heading=h.26in1rg)…………………………………………………………..30

**1.5.7.** [**Компаратив**](#_heading=h.lnxbz9)………………………………………………………32

**1.5.8.** [**Прислівник**](#_heading=h.35nkun2)……………………………………………………….33

**1.5.9.** [**Прийменник**](#_heading=h.44sinio)………………………………………………………36

**1.5.10.** [**Частка**](#_heading=h.2jxsxqh)…………………………………………………………….38

**1.5.11.** [**Предикатив**](#_heading=h.z337ya)………………………………………………………39

**1.5.12.** [**Дієприслівник**](#_heading=h.3j2qqm3)……………………………………………………41

[**РОЗДІЛ 2. Обчислення основних статистичних характеристик частин мови** 43](#_heading=h.1y810tw)

[**2.1 Іменник**](#_heading=h.2xcytpi)

[**2.2. Дієслово**](#_heading=h.1ci93xb)

[**2.3. Прикметник**](#_heading=h.3whwml4)

[**2.4. Прислівнк**](#_heading=h.2bn6wsx)

[**2.5. Числівник**](#_heading=h.qsh70q)

[**2.6. Займенник**](#_heading=h.3as4poj)

[**2.7 Прийменник**](#_heading=h.1pxezwc)

[**2.8 Сполучник**](#_heading=h.49x2ik5)

[**2.9 Частка**](#_heading=h.2p2csry)

**2.10** [**Дієприслівник**](#_heading=h.147n2zr)………………………………………………………………..

[**2.11 Предикатив**](#_heading=h.ihv636)

**2.12** [**Вигук**](#_heading=h.23ckvvd)…………………………………………………………………………..

**2.13** [**Компаратив**](#_heading=h.3o7alnk)…………………………………………………………………...

[**РОЗДІЛ 3. Зіставлення статистичних характеристик**](#_heading=h.32hioqz) 70

[**ВИСНОВКИ**](#_heading=h.1hmsyys)…………………………………………………………………….71

[**Додаток 1. Код для формування частотних словників за першим текстом** ……………………………………………………………………………………72](#_heading=h.41mghml)

[**Додаток 2. Код для обчислення статистичних характеристик**](#_heading=h.2grqrue)…………..83

# Використані джерела:

Програмний код для реалізації роботи був написаний на мові програмування Python3.7. В якості інтерпретатора було використано PyCharm - інтегроване середовище розробки для Python. В ході роботи було використано такі ресурси:

* Т.В. Орлова «Всесвітня історія. Історія цивілізацій».
* Валер’ян Підмогильний «Невеличка драма».
* sqlite3 – модуль бібліотеки полегшеної реляційної система керування базами даних SQLite.
* re – модуль, що забезпечує пошук регулярних виразів.
* pymorphy2 – морфологічний аналізатор.
* matplotlib - бібліотека на мові програмування Python для візуалізації даних двовимірною 2D графікою.
* <https://sqliteonline.com/> - інтернет сайт для створення, маніпулювання та перегляду баз даних.
* <https://regex101.com/> - інтернет сайт для пошуку регулярних виразів у тексті.
* <https://docs.python.org/3.7/> - документація Python 3.7.

# РОЗДІЛ 1. Створення електронного частотного словника

**Мета** цієї частини роботи – створити частотні словники словоформ, лем і частин мови за двома вибірками.

**Завдання**:

1. сформувати вибірки;
2. написати програмний код, що автоматично укладає частотні словники;
3. створити полігони частот частин мови.

## 1. Формування вибірок.

1.1.

Перша вибірка складена на основі тексту підручника Т.В. Орлової «Всесвітня історія. Історія цивілізацій». Вона належить до наукового стилю. Темою є історія стародавнього світу, історія людських цивілізацій.

Друга вибірка належить до художнього стилю. До неї включено уривок твору Валер’яна Підмогильного «Невеличка драма». За жанром це інтелектуальний роман. Головними темами роману є мікросвіт людини та її оточення, національна ідентичність, проблема українізації, конфлікт між розумом та почуттям, коханням та прагматизмом.

1.2.

Вибірки було сформовано зональним методом, що забезпечує їх лінгвістичну однорідність. Вони складаються з 20 тисяч слововживань. Кожна ділиться на 20 підвибірок по 1000 слововживань.

## 2. Укладання частотних словників

Другим етапом завдання було створення частотних словників за вибірками. Спочатку програмний код було написано для створення частотного словника словоформ.

Для цього за допомогою наступного фрагменту коду було створено базу даних *«frequency\_dict»,* а в ній таблицю *«част\_словоформ»* :

conn = sqlite3.connect(**'frequency\_dict.db'**)  
cursor = conn.cursor()  
  
cursor.execute(**'''CREATE TABLE IF NOT EXISTS част\_словоформ  
 (словоформа TEXT PRIMARY KEY NOT NULL,  
 gen\_freq INTEGER,  
 підв\_1 INTEGER,  
 підв\_2 INTEGER,  
 підв\_3 INTEGER,  
 підв\_4 INTEGER,  
 підв\_5 INTEGER,  
 підв\_6 INTEGER,  
 підв\_7 INTEGER,  
 підв\_8 INTEGER,  
 підв\_9 INTEGER,  
 підв\_10 INTEGER,  
 підв\_11 INTEGER,  
 підв\_12 INTEGER,  
 підв\_13 INTEGER,  
 підв\_14 INTEGER,  
 підв\_15 INTEGER,  
 підв\_16 INTEGER,  
 підв\_17 INTEGER,  
 підв\_18 INTEGER,  
 підв\_19 INTEGER,  
 підв\_20 INTEGER)  
'''**)

Вст. 1. Елемент коду, який створює таблицю «част\_словоформ».

Текст вибірки було переведено в формат txt для полегшення процесу подальшої роботи з ним. Оскільки наступний етап передбачає токенізацію слововживань та очищення тексту від зайвих символів, в результаті чого кількість слів у вибірці зменшиться, із самого початку було вибрано частину тексту на кількість слововживань більшу, ніж 20000, щоб за потреби не добирати слововживання вручну: програма автоматично відрахує потрібну кількість слів.

**with** open(**"вибірка 1.txt"**, encoding = **"utf-8"**) **as** data\_1:  
 text\_1 = data\_1.read().lower()

Вст. 2. Елемент коду, який зчитує текст вибірки у змінну text\_1 та переводить всі слова в нижній регістр.

Спочатку ми прибираємо з тексту всі зайві символи, а потім ділимо отриманий текст на окремі слововживання:

without\_punc\_marks = []  
edited = re.sub(**'\.'**, **''**, text\_1)  
edited = re.sub(**','**, **''**, edited)  
edited = re.sub(**'!'**, **''**, edited)  
edited = re.sub(**'\?'**, **''**, edited)  
edited = re.sub(**':'**, **''**, edited)  
edited = re.sub(**';'**, **''**, edited)  
edited = re.sub(**'\.\.'**, **''**, edited)  
edited = re.sub(**'\('**, **''**, edited)  
edited = re.sub(**'\)'**, **''**, edited)  
edited = re.sub(**'[a-zA-Z]+'**, **''**, edited)  
edited = re.sub(**'"'**, **''**, edited)  
edited = re.sub(**"'"**, **''**, edited)  
edited = re.sub(**'\['**, **''**, edited)  
edited = re.sub(**'\]'**, **''**, edited)  
edited = re.sub(**'\d'**, **''**, edited)  
edited = re.sub(**'%'**, **''**, edited)  
edited = re.sub(**'\\\\'**, **''**, edited)  
edited = re.sub(**'\n'**, **' '**, edited)  
edited = re.sub(**"'\]"**, **''**, edited)  
edited = re.sub(**"\['"**, **''**, edited)  
edited = re.sub(**'\s-\s'**, **' '**, edited)  
without\_punc\_marks.append(edited)

Вст. 3. Елемент коду, який прибирає з тексту зайві символи.

splitted\_1 = str(without\_punc\_marks).split(**' '**)

Вст. 4. Елемент коду, який здійснює токенізацію слововживань.

Усі слововживання ми присвоюємо змінній *splitted\_1* із типом даних «список». Останнім кроком в процесі редагування токенів є видалення «пустих» токенів зі списку *splitted\_1*.

Далі ми за допомогою циклу *for* відраховуємо зі *splitted\_1* потрібну кількість слововживань (20 тисяч). Код перебирає по черзі всі елементи списку слововживань (*splitted\_1*), рахує їх, додає до нового списку *edited\_list* і зупиняється, коли натрапляє на 20000-ий елемент:

count = 0  
edited\_list = []  
**for** word **in** splitted\_1:  
 **if** count == 20000:  
 **break** count += 1  
 edited\_list.append(word)

Вст. 5. Елемент коду, який відраховує зі списку слововживань 20 тисяч одиниць.

Після того, як вибірка сформована, необхідно поділити її на 20 підвибірок і обчислити в них кількість вживань кожної словоформи. Для цього зі списку елементів вибірки *edited\_list* було відраховано перші 1000 слів, так само, як це було зроблено на попередньому кроці для формування вибірки. Далі ми створюємо змінну *sample\_1\_dict* типу даних «словник», в яку в якості ключів будуть додаватися словоформи, а в якості значень – список із словоформи і її частот в усіх підвибірках. Тобто в результаті елементи словника виглядатимуть так: *«словоформа: «словоформа, частота в 1 підвибірці, частота в 2 підвибірці, …, частота в 20 підвибірці»».*

Процес обрахунку словоформ у першій підвибірці відбувається таким чином:

1. програма додає всі елементи, що входять до вибірки, до списку слововживань даної підвибірки (*sample\_1\_list*). Вони не потрібні для створення частотного словника словоформ, але необхідні пізніше для створення двох інших словників, щоб кожного разу не відраховувати по тисячі слів.
2. за умови, що словоформа відсутня в ключах словника, програма додає її до нього і записує до значення ключа цю словоформу і цифру 1. Словоформи ми додаємо до списку частот на нульову позицію для того, щоб потім одразу списком додати до таблиці в базі даних значення всіх колонок для кожної словоформи, а не додавати їх окремо від значень частот.
3. за умови, що словоформа вже існує в якості ключа словника, значення її частоти збільшується на один.

sample\_1\_dict = {}  
sample\_1\_list = []count = 0  
x = 0  
**for** word **in** edited\_list:  
 **if** count == 1000:  
 **break** count += 1  
 sample\_1\_list.append(word)  
 **if** word **in** sample\_1\_dict:  
 sample\_1\_dict[word][1] += 1  
 **else**:  
 sample\_1\_dict[word] = [word]   
 sample\_1\_dict[word].append(1)

Вст. 6. Елемент коду, який формує першу підвибірку і записує частоти всіх словоформ у ній.

Далі ми формуємо другу підвибірку. Її формування та обчислення частот словоформ відбувається за таким алгоритмом:

1. програма відбирає наступну тисячу слововживань зі списку *edited\_list*.
2. створюється пустий список, до якого цикл додає всі слововживання, що зустрічаються в ній (*sample\_2\_list)*.
3. за умови, що словоформи немає в ключах словника *sample\_1\_dict*, програма додає до значень на нульову позицію словофому, на першу – 0 (адже це означає, що в першій підвибірці її не було, і це необхідно зафіксувати), а на другу – 1.
4. якщо ж словоформа вже в ключах, то на другу позицію або додається 1 (якщо словоформа перший раз зустрілася в другій підвибірці), або ж значення другого елемента збільшується на один (якщо словоформа вже не вперше зустрілася в ній).

sample\_2\_list = []  
count = 0  
**for** word **in** edited\_list:  
 count +=1  
  
 **if** 1000 < count < 2001:  
 sample\_2\_list.append(word)  
 **if** word **in** sample\_1\_dict:  
 **try**:  
 sample\_1\_dict[word][2] += 1  
 **except**:  
 sample\_1\_dict[word].append(1)  
 **else**:  
 sample\_1\_dict[word] = [word]  
 x = 0  
 **for** i **in** sample\_1\_dict[word]:  
 **while** x < 1:  
 sample\_1\_dict[word].append(0)  
 x += 1  
 sample\_1\_dict[word].append(1)

Вст. 7. Елемент коду, який формує другу підвибірку і записує частоти всіх словоформ у ній.

Процедура додавання частот словоформ до значень словника у третій та всіх наступних підвибірках відбувається за одним алгоритмом, тому для цього було створено функцію *divide\_on\_samples*, яка в якості параметра приймає номер підвибірки.

Процедура дещо складніша, ніж в першій та другій підвибірках через те, що потрібно враховувати можливість, що словоформа в ключах є, але відсутня в якійсь кількості попередніх підвибірок. Тобто, наприклад, може бути таке, що словоформа зустрічалася в першій підвибірці, тому вона є в ключах словника, але її не було в другій. Тому нам необхідно додати до циклу команду, яка в такому випадку додаватиме 0 до значень на відповідних позиціях.

Функція, що обчислює кількість вживань кожної словоформи в 3-20 підвибірках працює таким чином:

1. приймає параметр – номер підвибірки.
2. якщо словоформа вже в ключах і не вперше зустрілася в даній підвибірці – програма збільшує значення її частоти на 1.
3. якщо словоформа в ключах, але зустрілася в даній підвибірці вперше – програма перевіряє кількість елементів в значеннях в *sample\_1\_dict*, приписаних до ключа з назвою цієї словоформи. Якщо їх кількість менша за номер підвибірки – словоформа була відсутня в одній або декількох попередніх.
4. додає потрібну кількість 0 на місця пропусків в значеннях словника *sample\_1\_dict.* Додає 1 – частоту словоформи в даній підвибірці.
5. якщо кількість елементів в значеннях в *sample\_1\_dict*, приписаних до ключа з назвою цієї словоформи дорівнює номеру підвибірки (тобто, словоформа була в усіх попередніх підвибірках) – програма просто додає 1.
6. якщо словоформи в ключах немає – тобто вона зустрілася вперше – додає ключ, до його значень додає словоформу, кількість нулів, на один меншу за номер підвибірки і 1.

**def** divide\_on\_samples(number):  
 **if** word **in** sample\_1\_dict:  
 **try**:  
 sample\_1\_dict[word][number] += 1  
 **except**:  
 sample\_len = len(sample\_1\_dict[word])  
 **if** sample\_len < number:  
 x = sample\_len  
 **while** x < number:  
 sample\_1\_dict[word].append(0)  
 x += 1  
 sample\_1\_dict[word].append(1)  
 **else**:  
 sample\_1\_dict[word].append(1)  
 **else**:  
 sample\_1\_dict[word] = [word]  
 x = 0  
 **while** x < number-1:  
 sample\_1\_dict[word].append(0)  
 x += 1  
 sample\_1\_dict[word].append(1)

Вст. 8. Елемент коду з функцією, записує частоти всіх словоформ у підвибірках, починаючи з третьої.

Перед викликом функції потрібно тільки сформувати підвібірку, тобто, відібрати наступну тисячу слів з *edited\_list*:

sample\_3\_list = []  
count = 0  
**for** word **in** edited\_list:  
 count +=1  
 **if** 2000 < count < 3001:  
 sample\_3\_list.append(word)  
 divide\_on\_samples(3)

Вст. 9. Елемент коду, який формує 3 підвибірку та викликає функцію, що додає частоти словоформ у вказаній підвибірці до словника sample\_1\_dict.

Після завершення формування підвибірок та обчислення частотності всіх словоформ в них необхідно відокремити значення словника *sample\_1\_dict* у список, що ми робимо за допомогою наступного елементу коду:

values1 = list(sample\_1\_dict.values())

Вст. 10. Елемент коду, який відокремлює значення словника в список.

Фактично ми отримуємо список списків, який присвоєно змінній *values1*. Потім за допомогою ітерації ми вставляємо до кожного елементу цього списку значення загальної частоти словоформи в вибірці на першу позицію (бо друга колонка в таблиці – це загальна частота). Загальну частоту ми отримуємо за допомогою методу *sum().* Він знаходить суму значень елементів списку з першої по двадцяту позицію, тобто суму частот в усіх підвибірках.

**for** i **in** values1:  
 gen\_freq = sum(i[1:21]) *#додаємо до списку загальну кількість слововживань* i.insert(1, gen\_freq)

Вст.11. Елемент коду, який обраховує суму частот у підвибірках і додає її в кожен елемент списку на першу позицію.

Потім розташовуємо одиниці частотного словника словоформ (*values1)* в порядку спадання їх загальної частоти, та присвоюємо значення відсортованого списку змінній *values1\_ordered*.

Програма сортує список за допомогою такого методу:

values1\_ordered = sorted(values1, key=**lambda** x:x[1], reverse=**True**)

Вст.12. Елемент коду, який розташовує елементи списку values1\_ordered за спаданням їх першого елемента (загальної частоти).

На цьому етапі всі потрібні дані знаходяться в *values1\_ordered*, тому їх можна додати до таблиці *«част\_словоформ»* і створити частотний словник. Ми робимо це за допомогою ітерації:

*#for i in values1\_ordered:  
# cursor.execute("""INSERT INTO част\_словоформ  
# VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)""", i)  
#conn.commit()*

Вст.13. Елемент коду, який додає всі дані до таблиці част\_словоформ.

Наступний етап завдання - заповнити таблицю *«част\_частин\_мови»,* тобто створюємо частотний словник частин мов у першій вибірці.

Для автоматичного визначення частин мови використовуємо модуль *pymorphy2*.

За допомогою наступної команди вибираємо мову, яку модуль аналізуватиме, тобто, українську:

morph = pymorphy2.MorphAnalyzer(lang=**'uk'**)

Вст.14. Елемент коду, який встановлює мову для аналізу.

А за допомогою наступних рядків коду ми вибираємо з усієї інформації, яку дає модуль *pymorphy2* про слово, його частиномовну приналежність:

parsed1 = morph.parse(i)[0]  
freq = parsed1.tag.POS

Вст.15. Елемент коду, який вибирає інформацію про частиномовну приналежність слів у змінну freq.

Знову ділити текст на підвибірки за допомогою циклів вже не потрібно, адже слововживання кожної підвибірки було записано в окремі списки *sample\_1\_list, sample\_2\_list* і т.д.

Тому за допомогою ітерації ми визначаємо частину мови кожного елемента в них та обраховуємо частоту частин мови. Останнє ми робимо за тим самим принципом, як і для словоформ, з тією лише різницею, що змінна типу даних «словник», в яку ми зберігаємо всі значення має іншу назву – *«part\_of\_speech»,* а функція, яка рахує частини мови для всіх підвибірок, починаючи з третьої – *count\_parts\_of\_speech*.

part\_of\_speech = {}

**for** i **in** sample\_1\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS**if** freq **in** part\_of\_speech:  
 part\_of\_speech[freq][1] += 1  
 **else**:  
 part\_of\_speech[freq] = [freq]  
 part\_of\_speech[freq].append(1)

**for** i **in** sample\_2\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS  
 *#print(freq)* **if** freq **in** part\_of\_speech:  
 **try**:  
 part\_of\_speech[freq][2] += 1  
 **except**:  
 part\_of\_speech[freq].append(1)  
 **else**:  
 part\_of\_speech[freq] = [freq]  
 x = 0  
 **for** o **in** part\_of\_speech[freq]:  
 **while** x < 1:  
 part\_of\_speech[freq].append(0)  
 x += 1  
 part\_of\_speech[freq].append(1)**def** count\_parts\_of\_speech(number):  
 **if** freq **in** part\_of\_speech:  
 **try**:  
 part\_of\_speech[freq][number] += 1  
 **except**:  
 freq\_len = len(part\_of\_speech[freq])  
 **if** freq\_len < number:  
 x = freq\_len  
 **while** x < number:  
 part\_of\_speech[freq].append(0)  
 x += 1  
 part\_of\_speech[freq].append(1)  
 **else**:  
 part\_of\_speech[freq].append(1)  
 **else**:  
 part\_of\_speech[freq] = [freq]  
 x = 0  
 **while** x < number-1:  
 part\_of\_speech[freq].append(0)  
 x += 1  
 part\_of\_speech[freq].append(1)  
  
**for** i **in** sample\_3\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POScount\_parts\_of\_speech(3)

Вст.16. Елемент коду, який визначає та рахує частини мови словоформ в першій, другій та третій підвибірках.

Лематизацію проводимо за допомогою того самого модуля - *pymorphy2*. Команда *normal\_form* зводить слово до початкової форми.

Процес обчислення частот лем у кожній підвибірці відбувається так само, як і обчислення частот частин мови, тільки тепер всі значення записуються у змінну *lemmas*, а функція має назву *count\_lemmas*:

lemmas = {}  
**for** i **in** sample\_1\_list:  
 normal\_form = morph.parse(i)[0].normal\_form**if** normal\_form **in** lemmas:  
 lemmas[normal\_form][1] += 1  
 **else**:  
 lemmas[normal\_form] = [normal\_form]  
 lemmas[normal\_form].append(1)

**for** i **in** sample\_2\_list:  
 normal\_form = morph.parse(i)[0].normal\_form  
 **if** normal\_form **in** lemmas:  
 **try**:  
 lemmas[normal\_form][2] += 1  
 **except**:  
 lemmas[normal\_form].append(1)  
 **else**:  
 lemmas[normal\_form] = [normal\_form]  
 x = 0  
 **for** o **in** lemmas[normal\_form]:  
 **while** x < 1:  
 lemmas[normal\_form].append(0)  
 x += 1  
 lemmas[normal\_form].append(1)**def** count\_lemmas(number):  
 **if** normal\_form **in** lemmas:  
 **try**:  
 lemmas[normal\_form][number] += 1  
 **except**:  
 lemmas\_len = len(lemmas[normal\_form])  
 **if** lemmas\_len < number:  
 x = lemmas\_len  
 **while** x < number:  
 lemmas[normal\_form].append(0)  
 x += 1  
 lemmas[normal\_form].append(1)  
 **else**:  
 lemmas[normal\_form].append(1)  
 **else**:  
 lemmas[normal\_form] = [normal\_form]  
 x = 0  
 **for** o **in** lemmas[normal\_form]:  
 **while** x < number-1:  
 lemmas[normal\_form].append(0)  
 x += 1  
 lemmas[normal\_form].append(1)  
  
**for** i **in** sample\_3\_list:  
 normal\_form = morph.parse(i)[0].normal\_form  
 count\_lemmas(3)

Вст.17. Елемент коду, який ставить слова в першій вибірці в початкову форму та рахує їх в межах даної вибірки.

В результаті ми отримуємо по три таблиці в базі даних на кожну вибірку. Це – частотні словники словоформ, частин мови і лем.

Складання частотних словників за другою підвибіркою відбувається так само, тільки таблиці в базі даних мають інші назви: *«част\_словоформ\_2»* , *«част\_частин\_мови\_2»* і *«част\_лем\_2»*.

## 3. Частотні словники

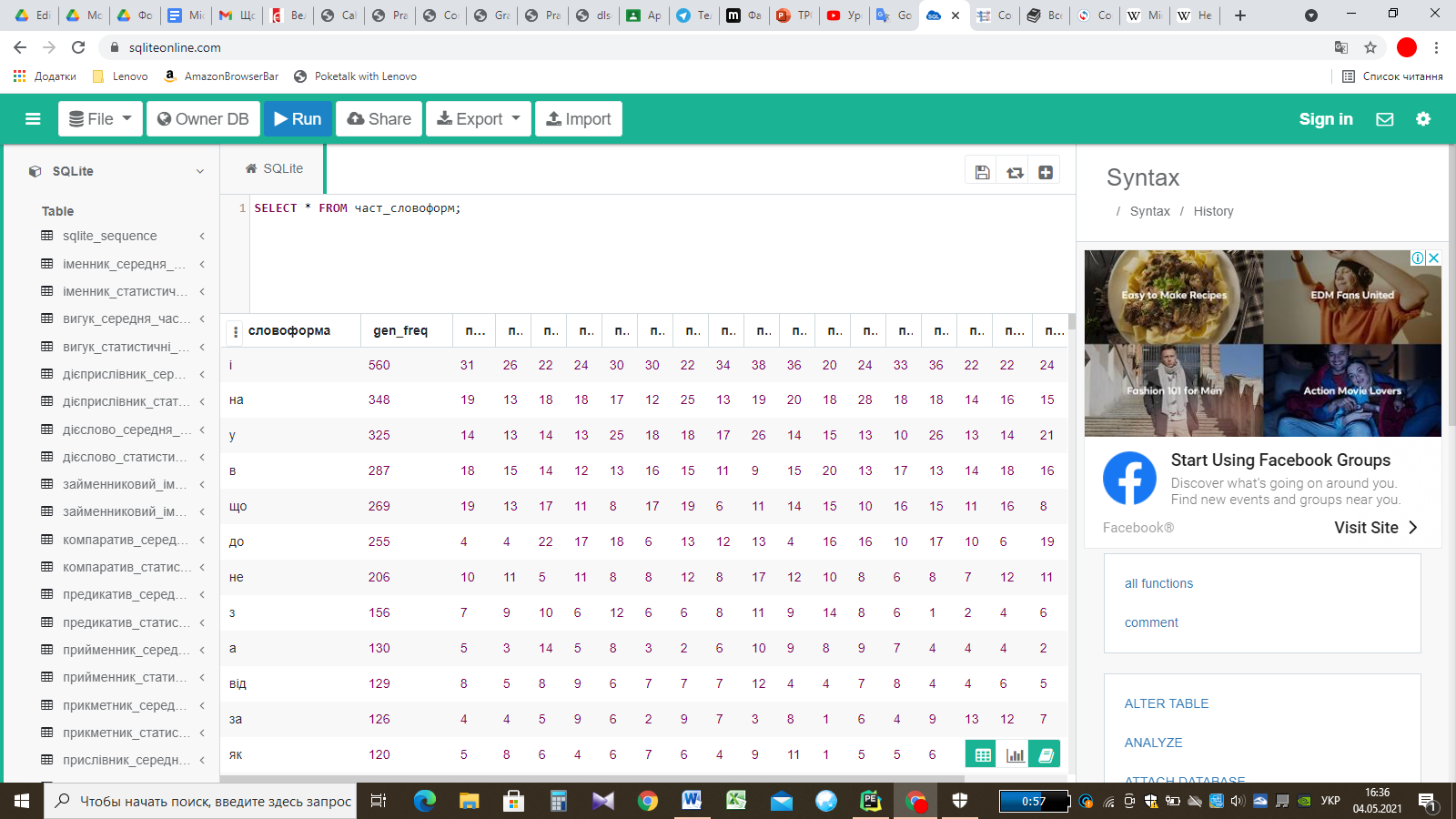
Таблиці *«част\_словоформ»* і *«част\_словоформ\_2*» складаються з наступних стовпців: словоформа, абсолютна частота, варіаційний ряд. Виглядають вони таким чином:

Рис. 1. Таблиця частотного словника словоформ в першій вибірці.

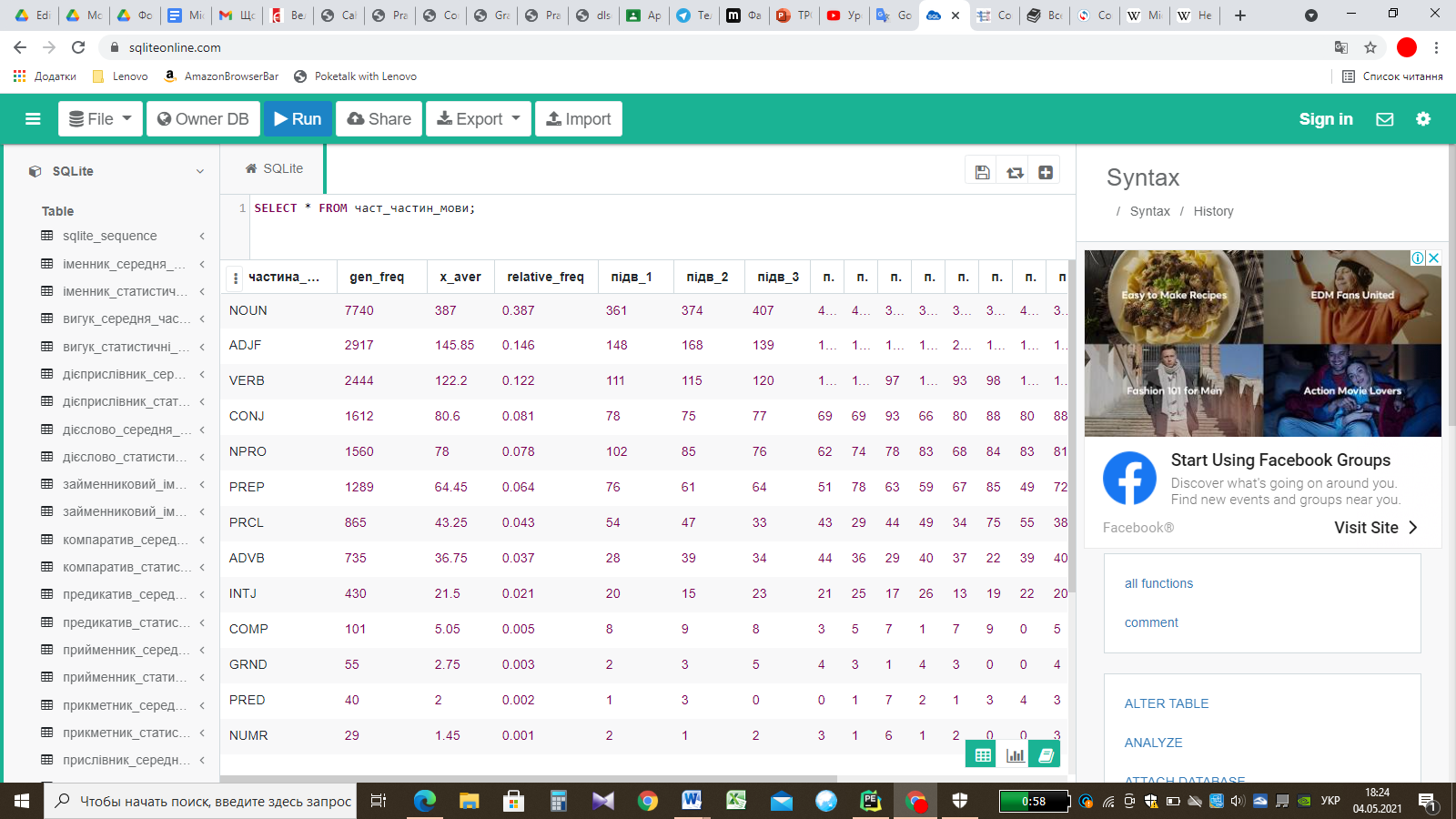
Таблиці *«част\_частин\_мови»* і *«част\_частин\_мови\_2»* мають таку структуру: назва частини мови, абсолютна частота, середня частота, відносна частота, варіаційний ряд. Так виглядає «част\_частин\_мови»:

Рис. 2. Таблиця частотного словника частин мови в першій вибірці.

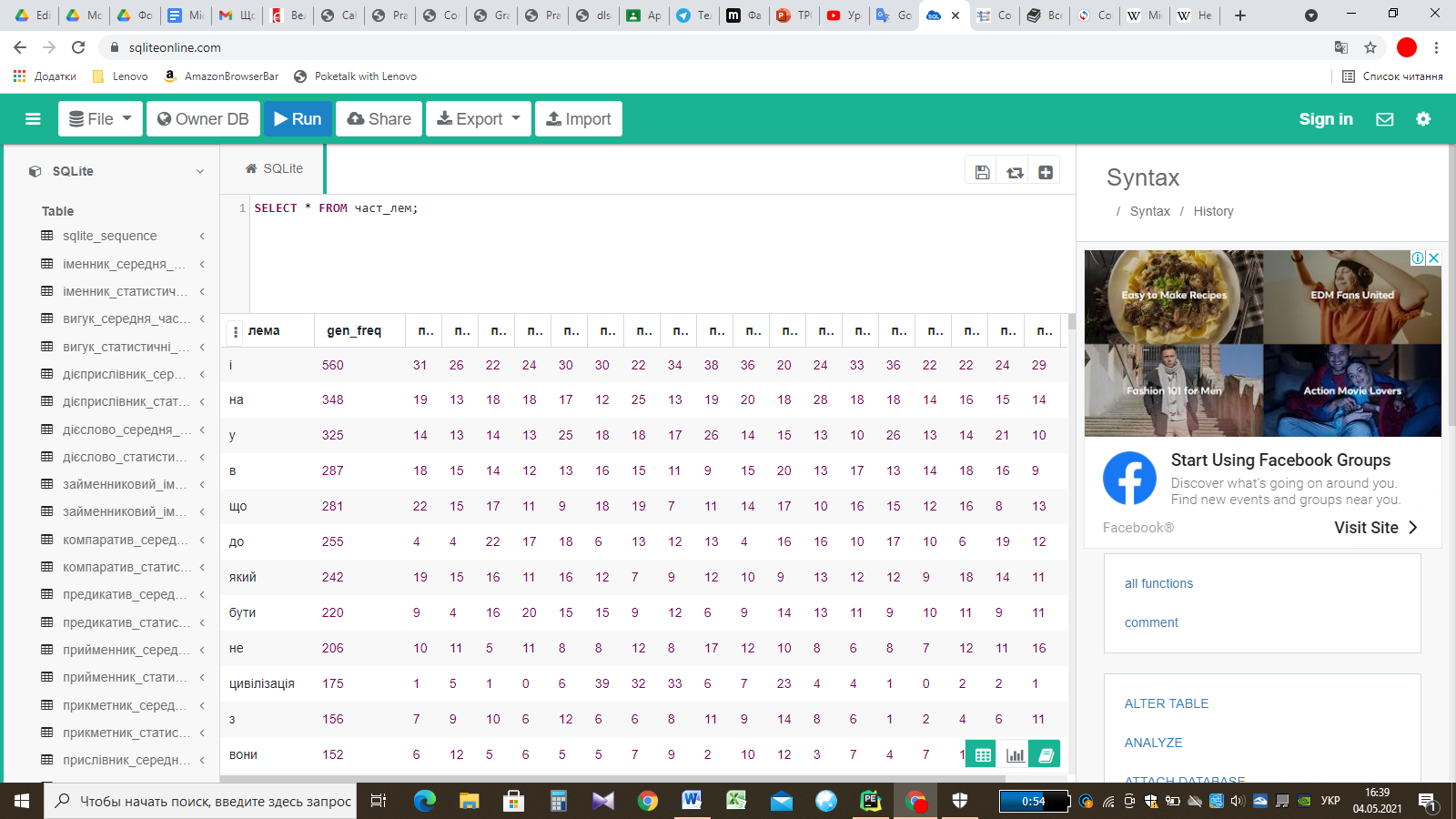
Структура таблиць *«част\_лем»* і *«част\_лем\_2»* наступна: лема і варіаційний ряд частот. Таблиці мають такий вигляд:

Рис. 3. Таблиця частотного словника лем в першій вибірці.

В усіх таблицях дані впорядковані за спаданням абсолютної частоти одиниць.

Як зазначалося вище, всі частотні словники знаходяться в одній базі даних, але вони не пов’язані між собою ніякими зовнішніми ключами.

## 4. Характеристика вибірок.

У першій вибірці найчастотнішим словом є «і», його абсолютна частота становить 560 вживань. За таблицею залежності кількості прикладів від заданої відносної похибки встановлюємо, що його відносна похибка – 8.3%.

Другим за частотністю словом у цій вибірці є «на», абсолютна частота якого – 348 вживань. За таблицею залежності кількості прикладів від заданої відносної похибки маємо, що його відносна похибка – 15%.

Низько частотними словами є, наприклад, слово «жінка» (6 вживань) і «всесвіт» (3 вживання). Відносна похибка для першого становить 80%, а для другого – 114%.

Для того, аби їх відносна похибка була 5 %, необхідно мати 1538 вживань цих слів. Тому для статистичного дослідження першого слова обсяг вибірки має бути приблизно 5 126 667 слововживань, а для другого – 10 253 334 слововживання.

У другій вибірці найчастотнішим є слово «не» з абсолютною частотою 398. За тією ж таблицею встановлюємо, що його похибка становить 9.8%.

Друге за частотністю слово в даній вибірці – це «і», абсолютна частота якого дорівнює 367 вживанням, а похибка за тією ж таблицею становить 13%.

До низькочастотних слів належать «жінка» (9 вживань) і «всіх» (5 вживань). Оскільки для відносної похибки у 5% необхідно мати 1538 прикладів вживання одиниці, то для статистичного дослідження першого слова обсяг вибірки має бути приблизно 3 417 778 слововживань, а для другого – 6 152 000 слововживання.

Проаналізуємо абсолютні частоти частин мови в обох вибірках на достатність для статистичного дослідження. Для цього будемо спиратися на таблицю залежності кількості прикладів від заданої відносної похибки.

Якщо вважати допустимою похибку у 20%, то в першій вибірці обсяг вибірки є достатнім для іменника (відносна похибка близько 2%), прикметника (3.6%), дієслова (близько 4%), сполучника (майже 5%), займенникового іменника (5%), прийменника (приблизно 5.5%), частки (приблизно 6.7%), прислівника (7%) та вигука (9.5%).

У другій вибірці за допустимої похибки у 20% обсяг вибірки є достатнім для статистичного дослідження таких частин мови: іменник (близько 2.7%), дієслово (3.4%), займенниковий іменник (близько 3.8%), прикметник (близько 4.4%), частка (4.8%), сполучник (майже 5%), прислівник (5.2%), прийменник (приблизно 6.1%), вигук (9.8%) і дієприслівник (14.3%).

## 5. Полігони частот.

Наступним етапом роботи була автоматична побудова полігону частот для кожної частини мови в обох підвибірках. Цей етап було реалізовано в другій частині лабораторної роботи (Додаток 2).

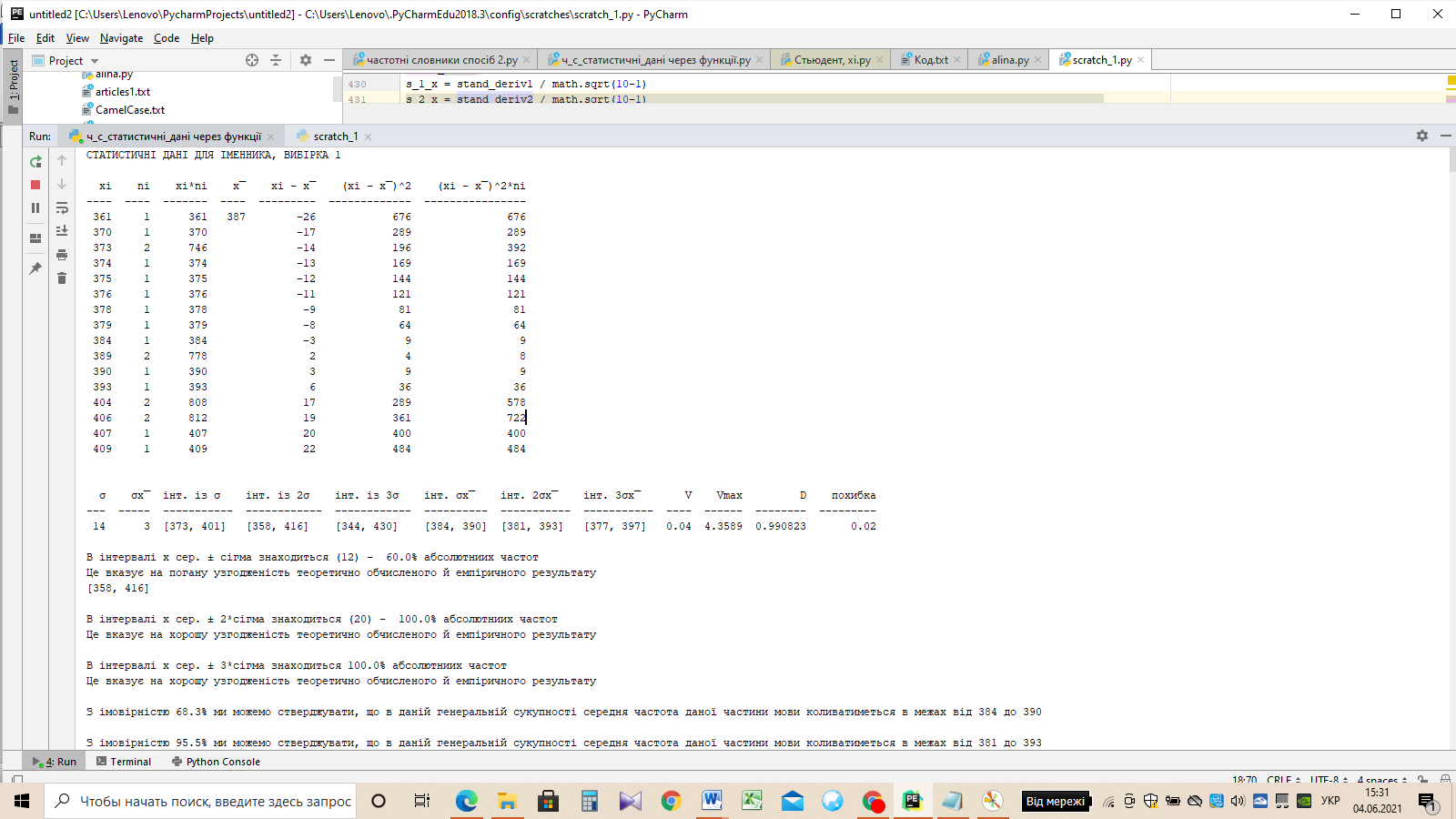
Для побудови графіків було використано бібліотеку *matplotlib*. Полігони частот однієї частини мови в різних підвибірках було поміщено на один графік для зручності порівняння. Частина коду, що автоматично його будує виглядає так:

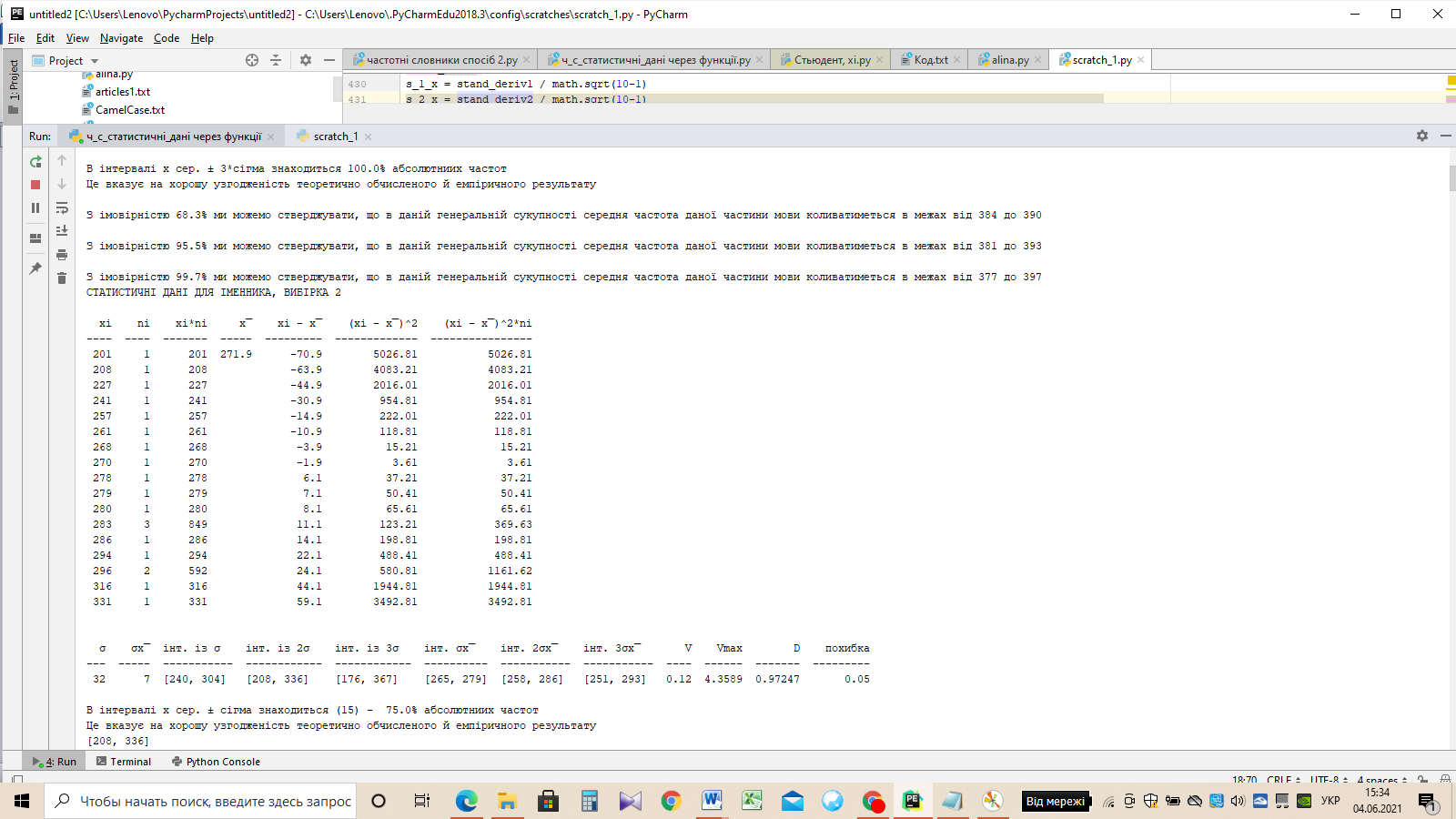
plt.plot(xi, ni, label = **'1 вибірка'**, marker = **'o'**, , markersize = 4)  
plt.plot(xi, ni, label = **'2 вибірка'**, marker = **'o'**, markersize = 4)  
plt.xlabel(**'xi'**)  
plt.ylabel(**'ni'**)  
plt.title(**'полігон частот іменника'**)  
plt.legend()  
plt.show()

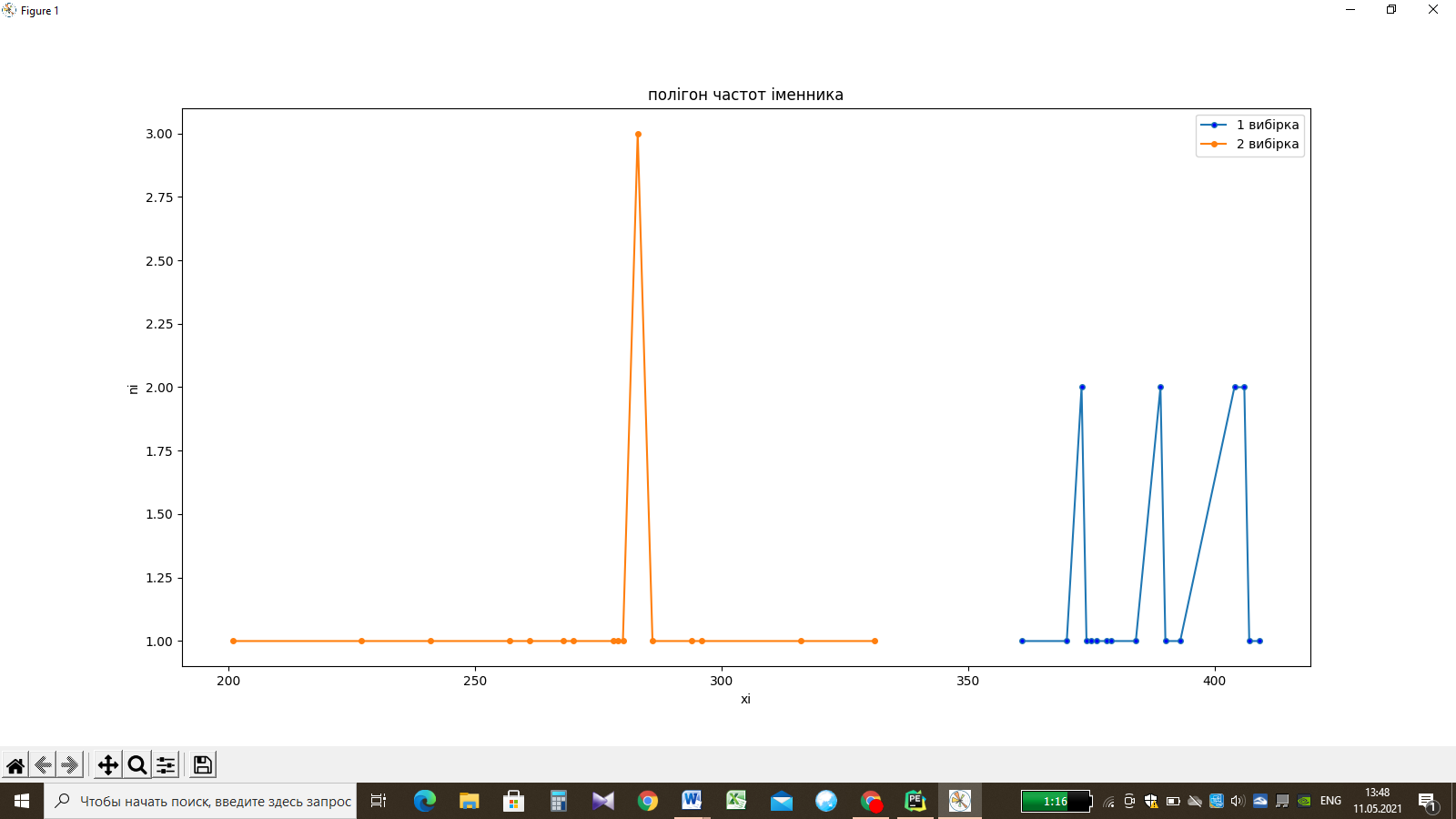
Вст.14. Елемент коду, який за допомогою модулю matplotlib створює графік полігонів частот іменника в обох вибірках.

хі– це списки абсолютних частот у вибірках, а ni– списки їх кількостей.

### Іменник

Варіаційний ряд іменника:

Полігон частот іменника: 

Як видно на графіку, абсолютні частоти іменника в першій вибірці коливаються в менших межах (від 361 до 409), ніж в другій (від 201 до 331), і лише чотири з них повторюються двічі. У другій вибірці всі, крім однієї, абсолютні частоти зустрічається один раз. Значення абсолютних частот у вибірках не перетинаються. Іменник зустрічається частіше в першій вибірці, в ній його середня частота майже на 100 вища.

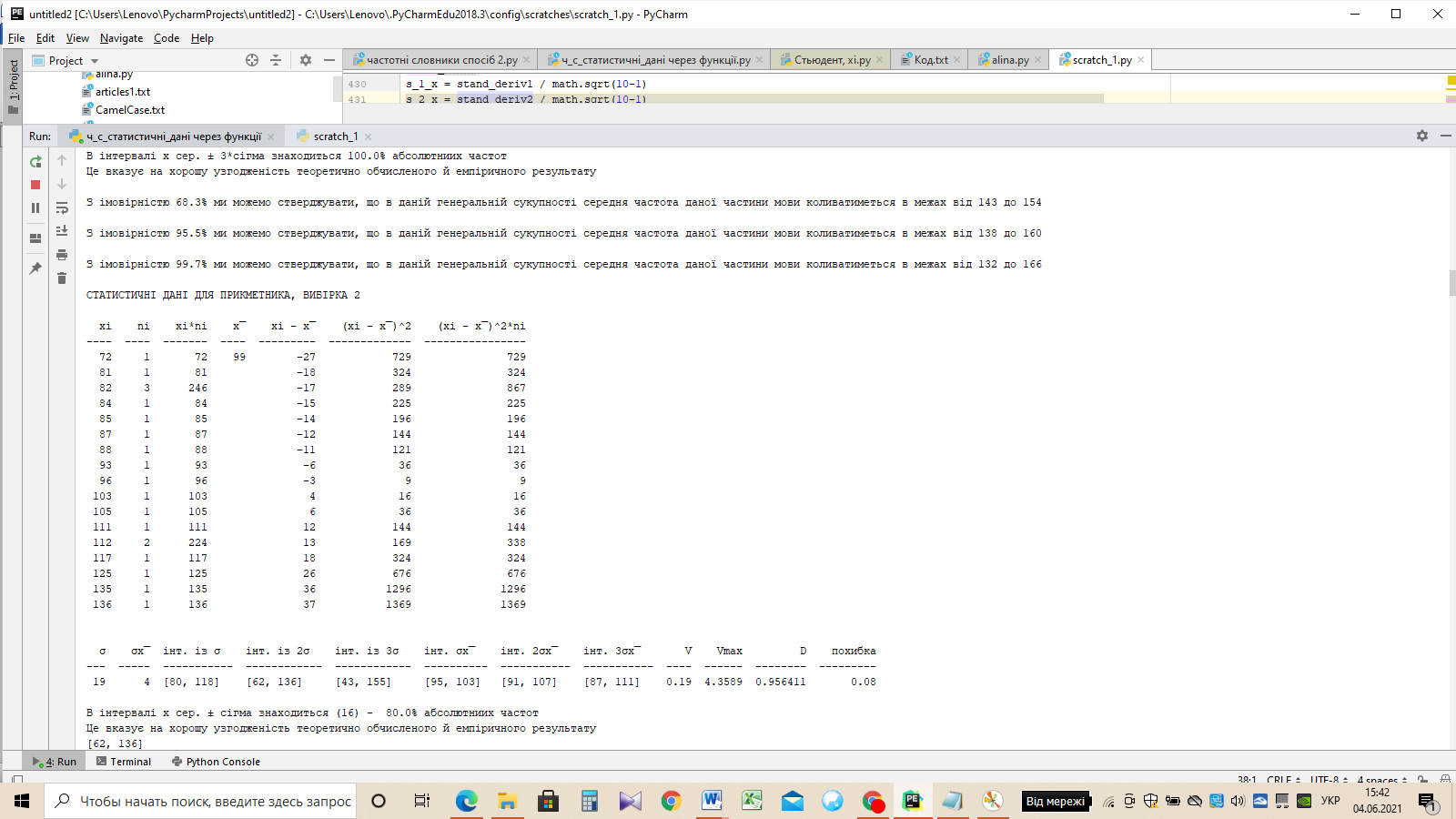
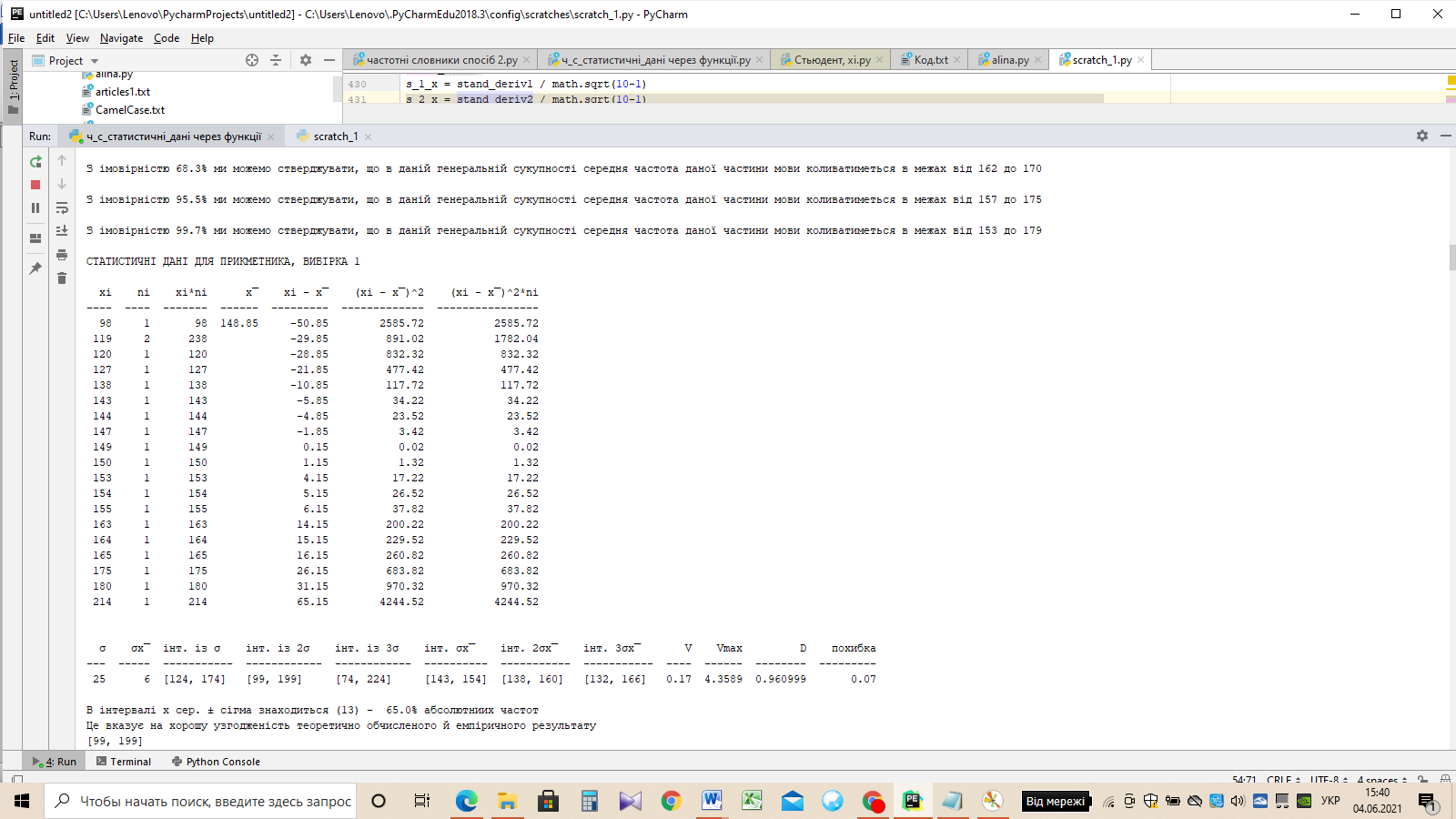
Але такий графік не відображає особливості розподілу абсолютних частот. Тому побудуємо графік полігонів частот за інтервальним. Він має такий вигляд:

Тепер абсолютні частоти згруповані навколо одного центру, і легше зробити висновок про їх розподіл. В першій вибірці найбільше абсолютних частот знаходиться в інтервалі з центром 375 (7), і також багато абсолютних частот припадає на інтервал з центром 400 (6). Беручи до уваги цей факт і те, що різниця міє мінімальним і максимальним значенням абсолютної частоти не велика, можна зробити висновок, що в першій вибірці іменник розподілений доволі рівномірно.

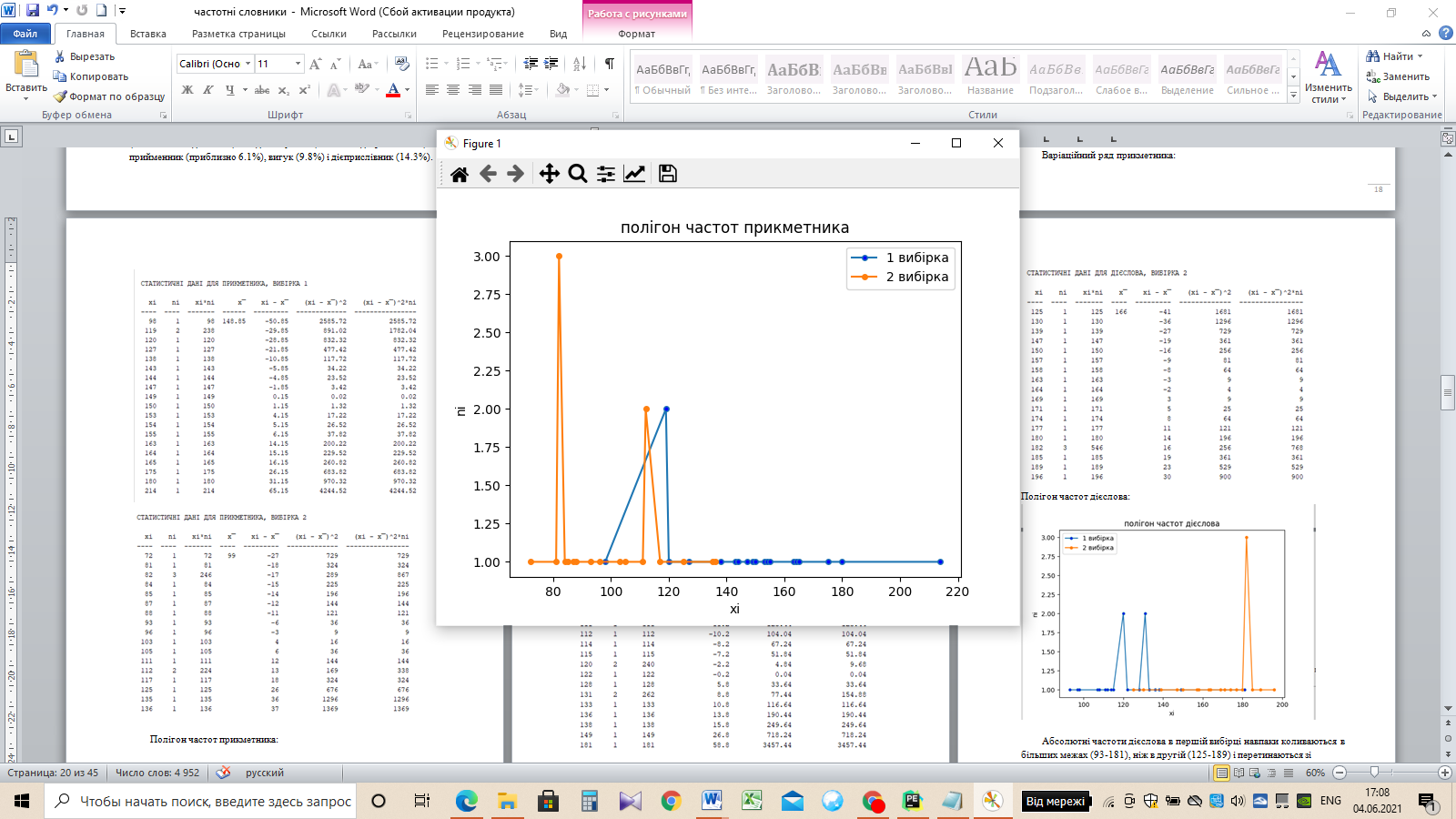
В другій вибірці найбільше абсолютних частот знаходиться в інтервалі з центром 280 (6). В інших інтервалах знаходиться від 0 до 3 абсолютних частот. Беручи до уваги цей факт і те, що різниця міє мінімальним і максимальним значенням абсолютної частоти доволі значна, можна зробити висновок, що в другій вибірці іменник розподілений менш рівномірно, ніж в першій.

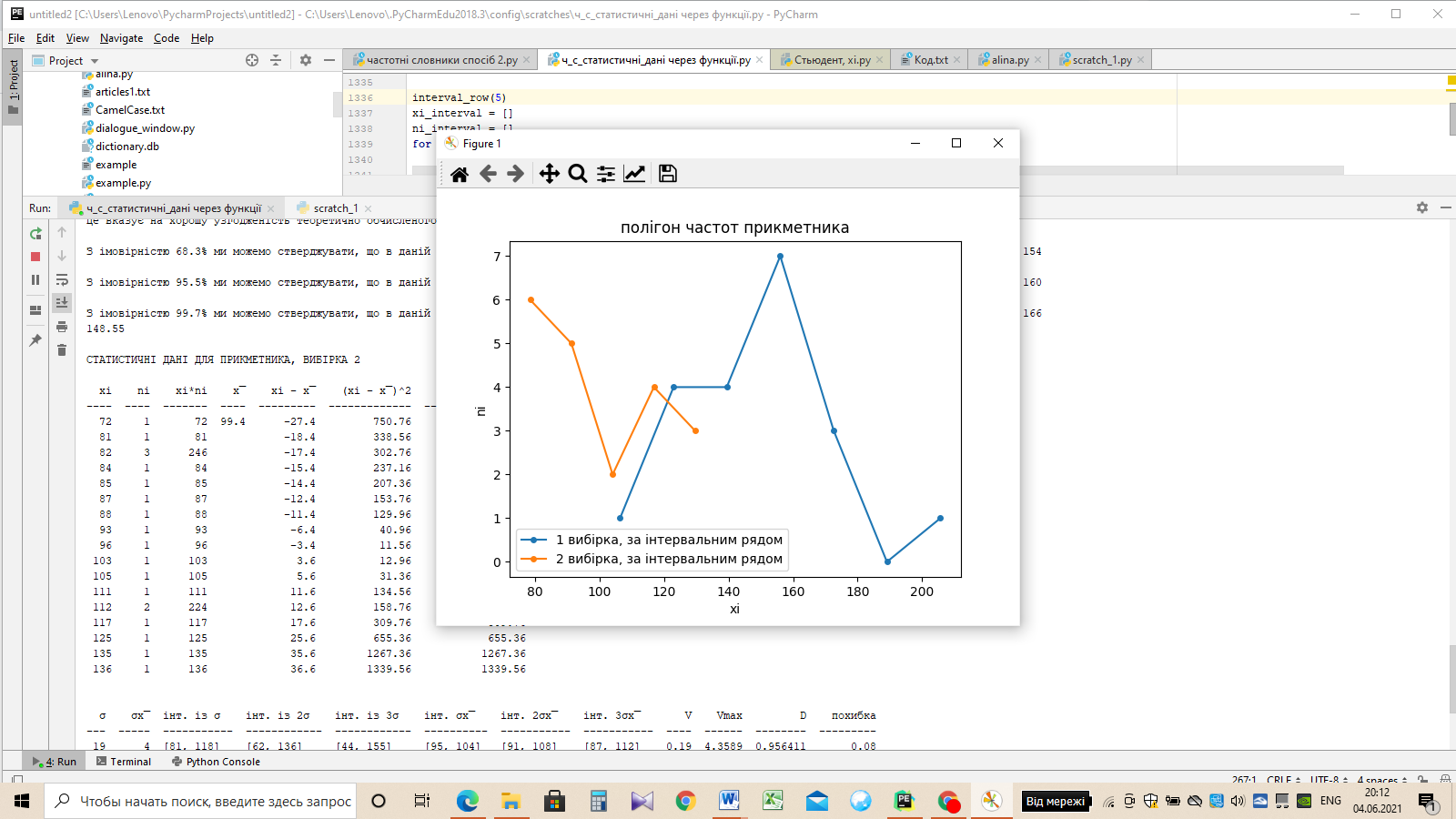
### Прикметник

Варіаційний ряд прикметника:



Полігон частот прикметника:

Абсолютні частоти прикметника в першій вибірці знаходяться у проміжку від 96 до 210. А в другій в проміжку від 72 до 136. Їх значення перетинаються, але середнє значення частоти в першій вибірці значно більше, ніж в другій. Також можна зробити висновок, що прикметник вжавається в першій вибірці частіше, ніж в другій.

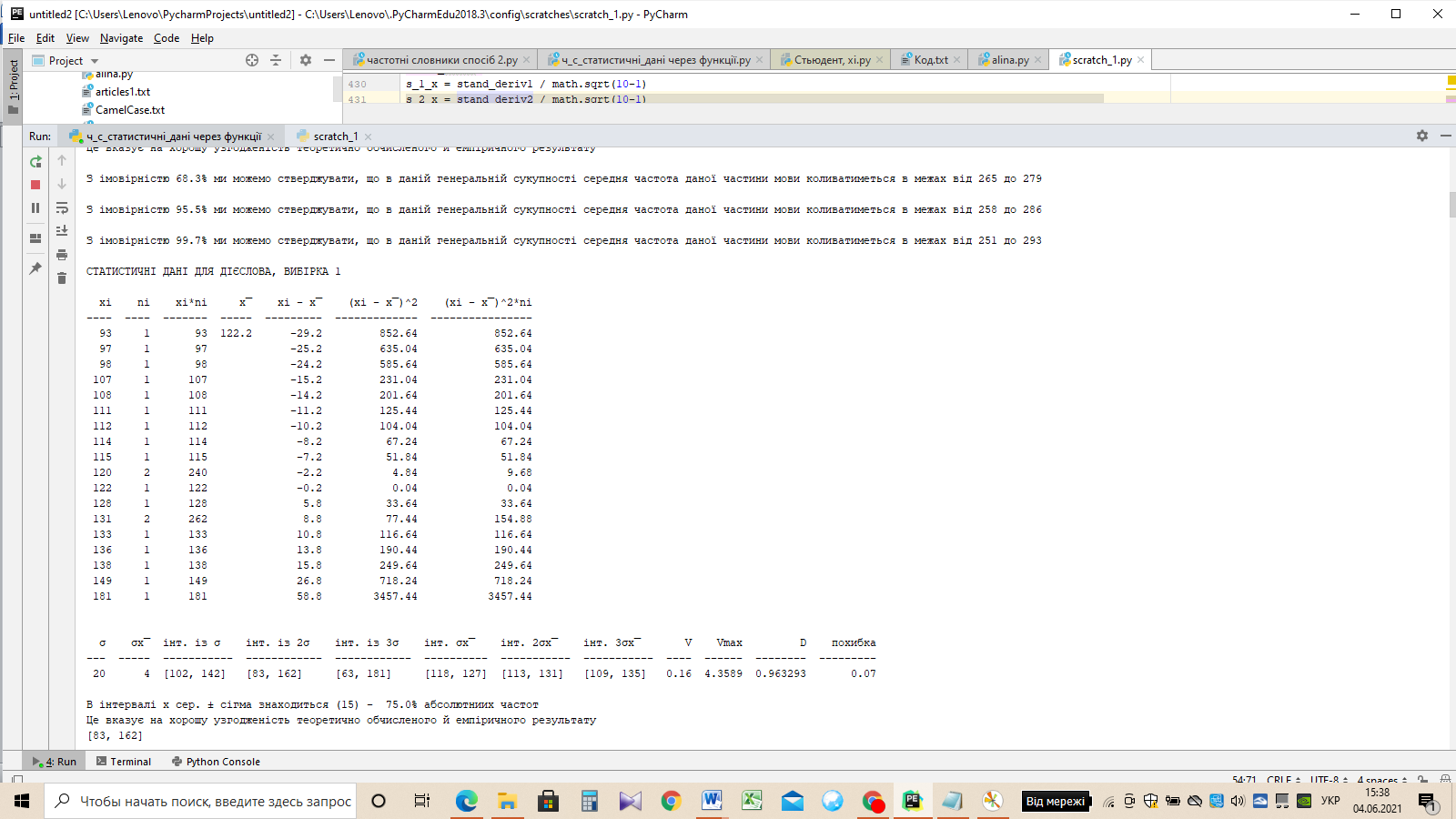
Але побачити особливості групування абсолютних частот на цьому графіку полігонів частот також складно, тому побудуймо полігони частот за інтервальними рядами:

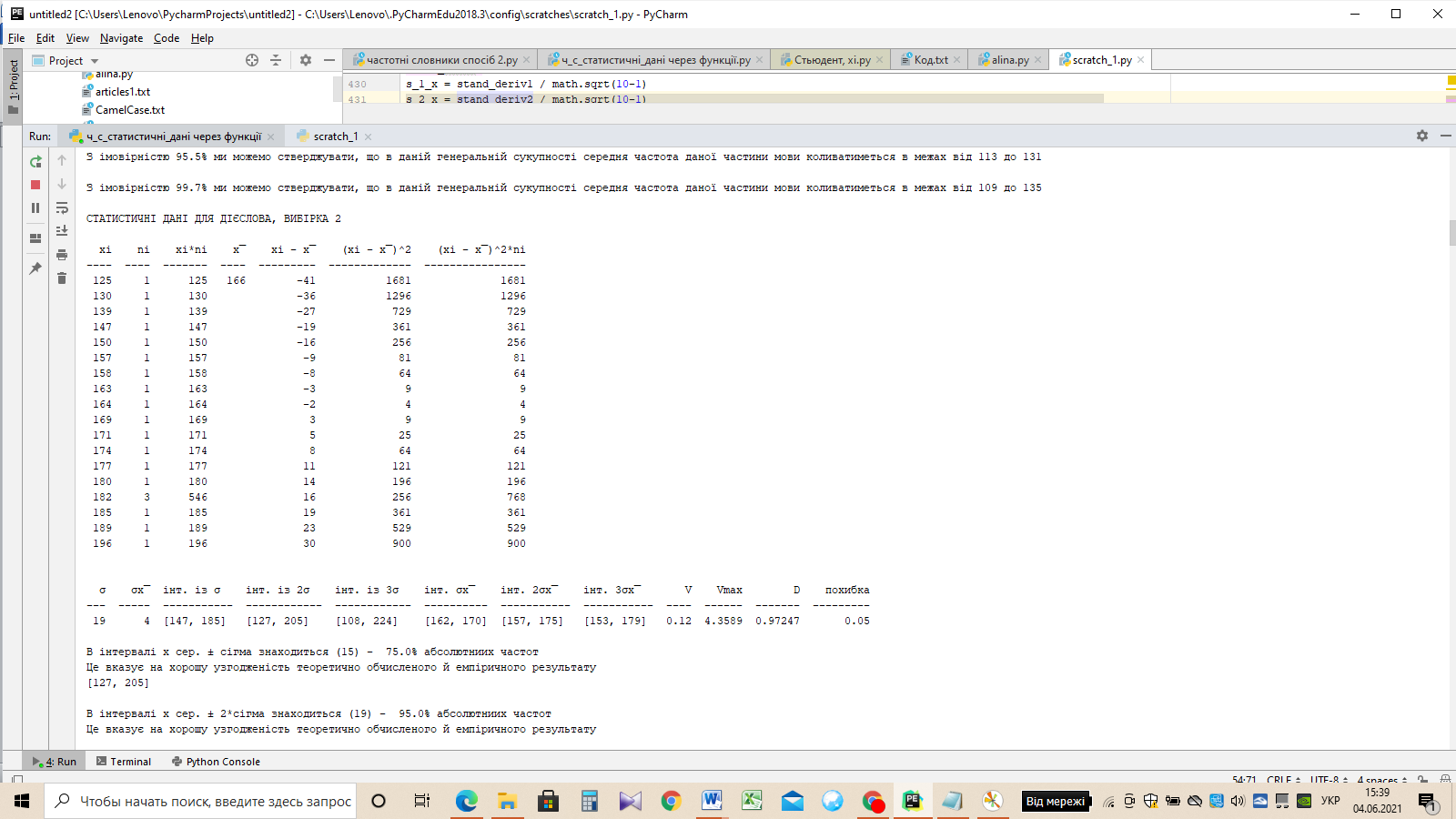
Тепер абсолютні частоти згруповані навколо одного центру, і легше зробити висновок про їх розподіл. В першій вибірці найбільше абсолютних частот знаходиться в інтервалі з центром 158 (7). Беручи до уваги цей факт і те, що різниця міє мінімальним і максимальним значенням абсолютної частоти не велика, можна зробити висновок, що в першій вибірці прикметник розподілений доволі рівномірно.

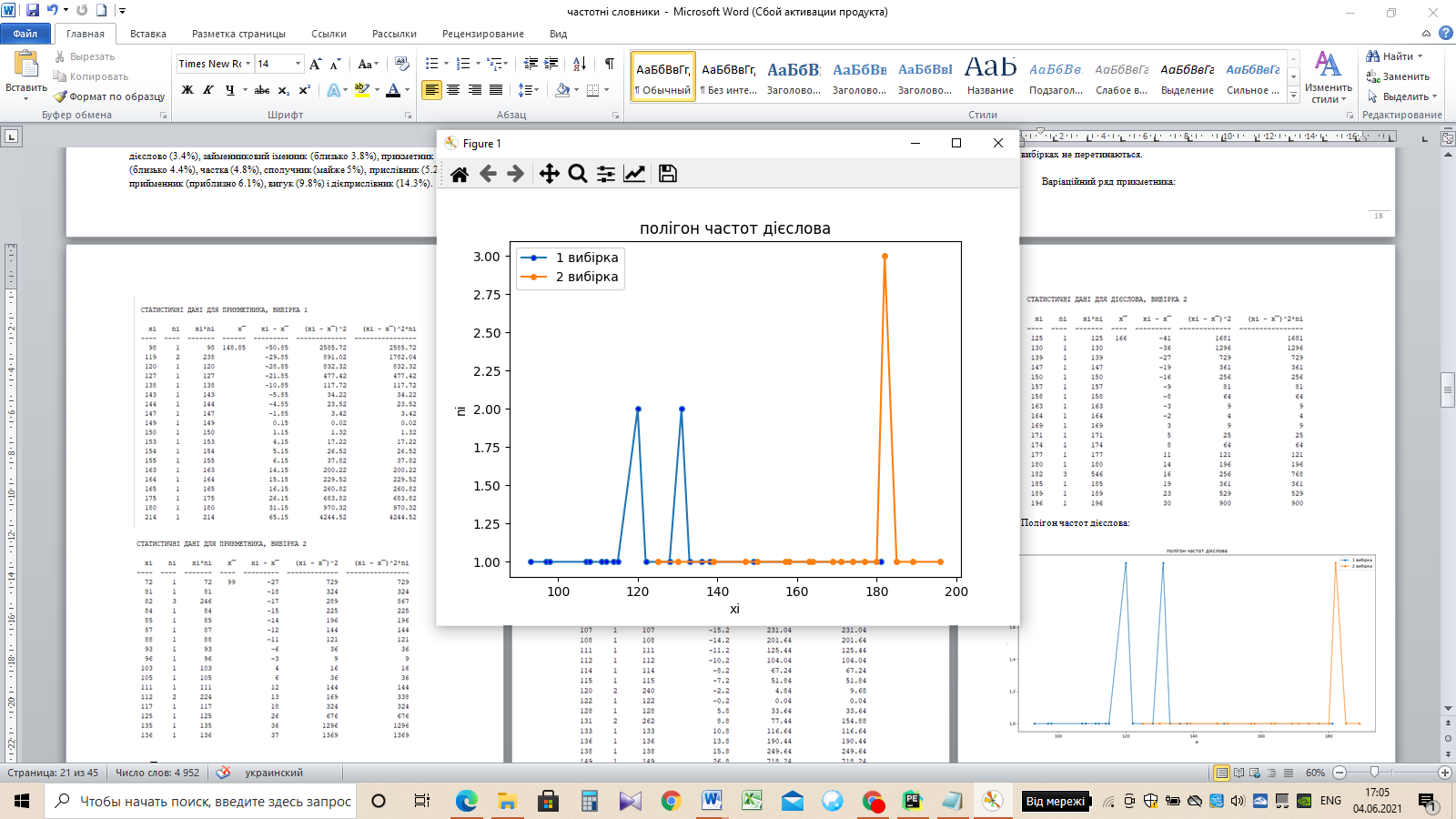
В другій вибірці найбільше абсолютних частот знаходиться в інтервалі з центром 80 (6), а також багато абсолютних частот припадає на інтервал з центрм 90(5). 4 значення абсолютної частоти знаходяться в інтервалі з центрм 120. Беручи до уваги цей факт і те, що різниця міє мінімальним і максимальним значенням абсолютної частоти не значна, можна зробити висновок, що в другій вибірці прикметник розподілений рівномірно.

### Дієслово

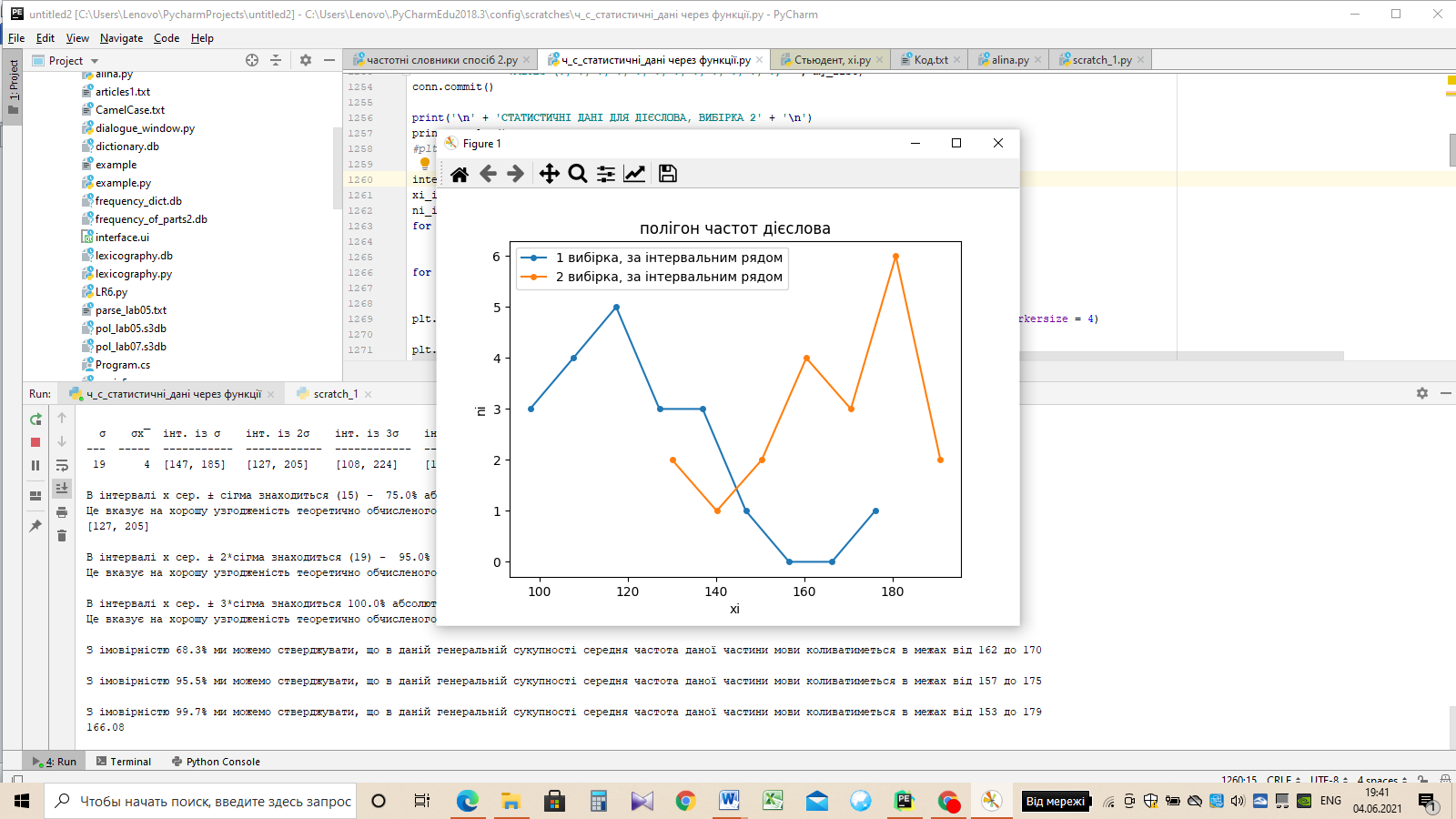
Варіаційний ряд дієслова:



Полігон частот дієслова:

Абсолютні частоти дієслова в першій вибірці навпаки коливаються в більших межах (93-181), ніж в другій (125-189) і перетинаються зі значеннями абсолютних частот в другій вибірці. Майже всі значення в обох вибірках зустрічаються по разу. На відміну від попередніх частин мови, дієслово часттіше вживається в другій вибірці.

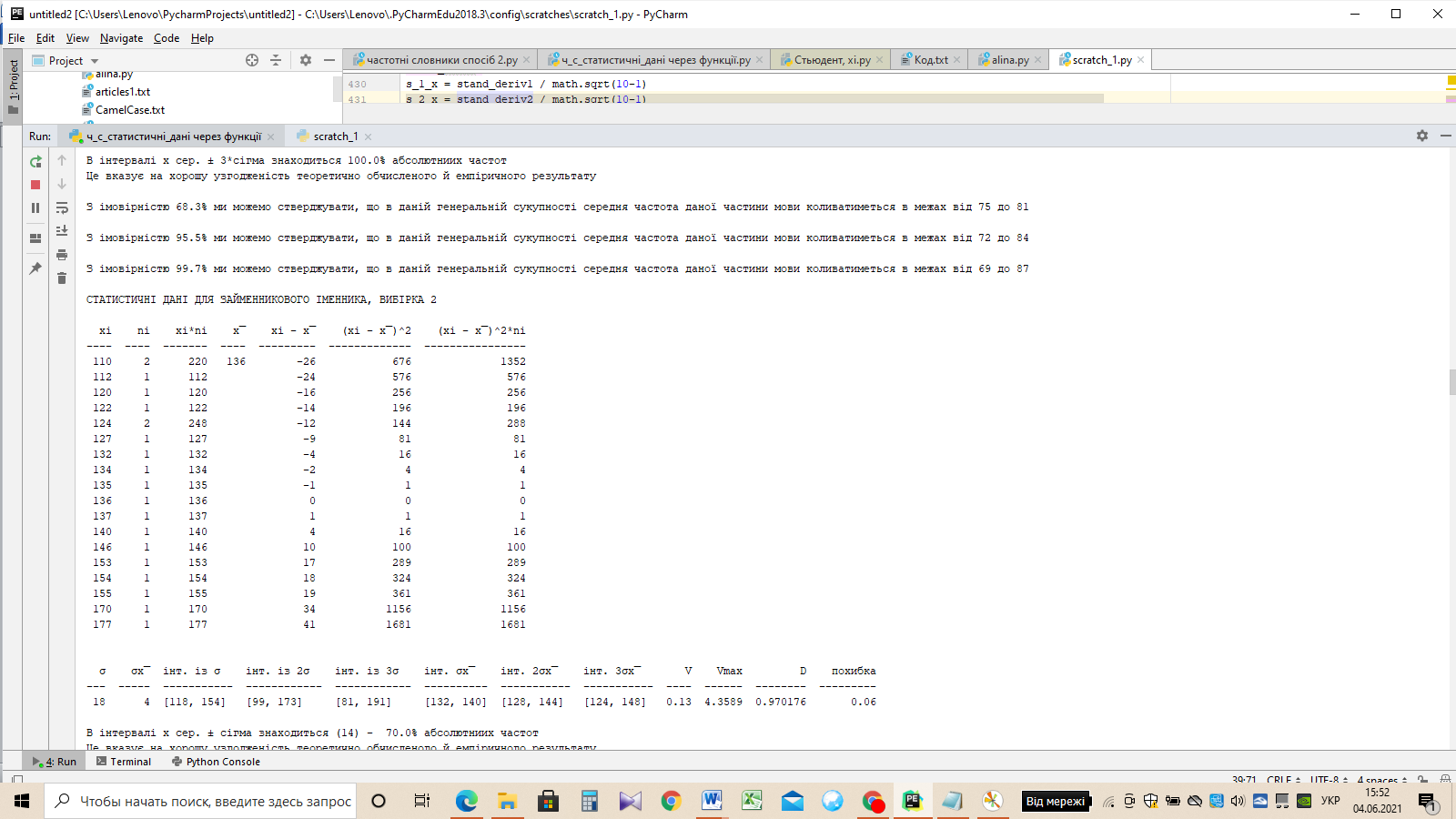
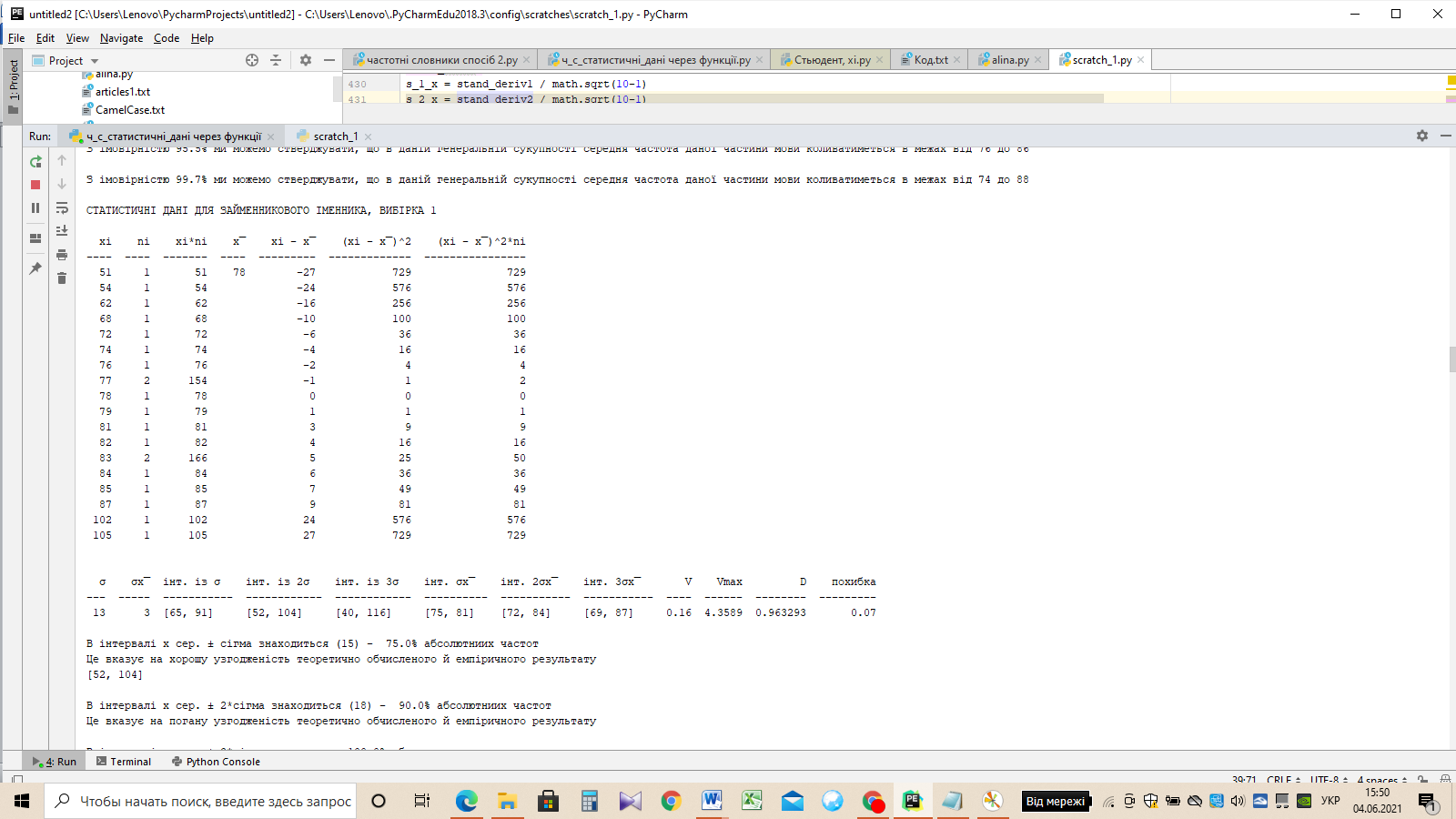
Полігони частот не відображають особливості розподілу абсолютних частот, тому варто побудувати полігони частот за інтервальними рядами:



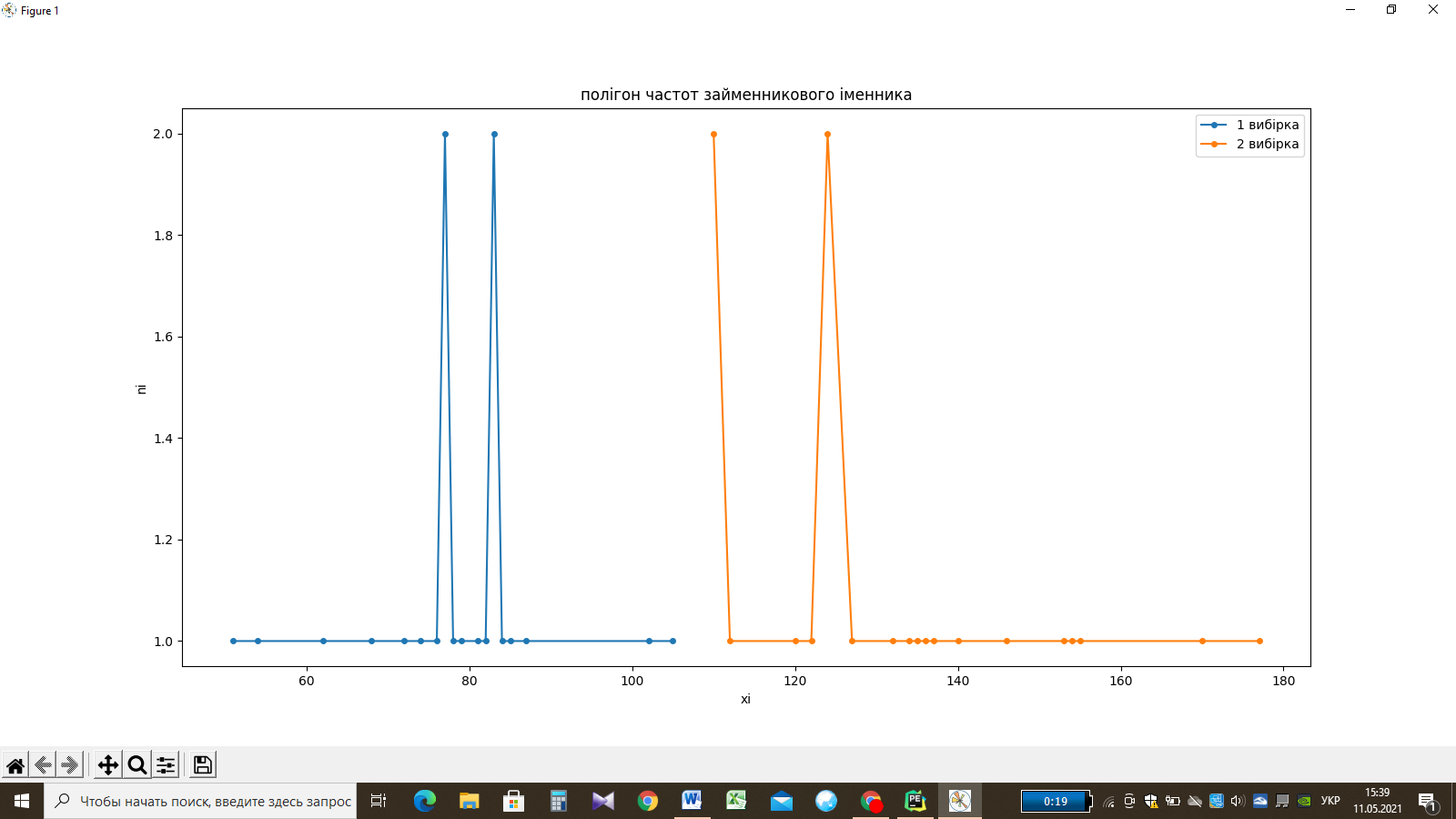
Тепер абсолютні частоти згруповані навколо одного центру, і легше зробити висновок про їх розподіл. В першій вибірці найбільше абсолютних частот знаходиться в інтервалі з центром 120 (5). На проміжок з центром 110 припадає 4 абсолютні частоти. Беручи до уваги цей факт і те, що різниця міє мінімальним і максимальним значенням абсолютної частоти не велика, можна зробити висновок, що в першій вибірці дієслово розподілено доволі рівномірно.

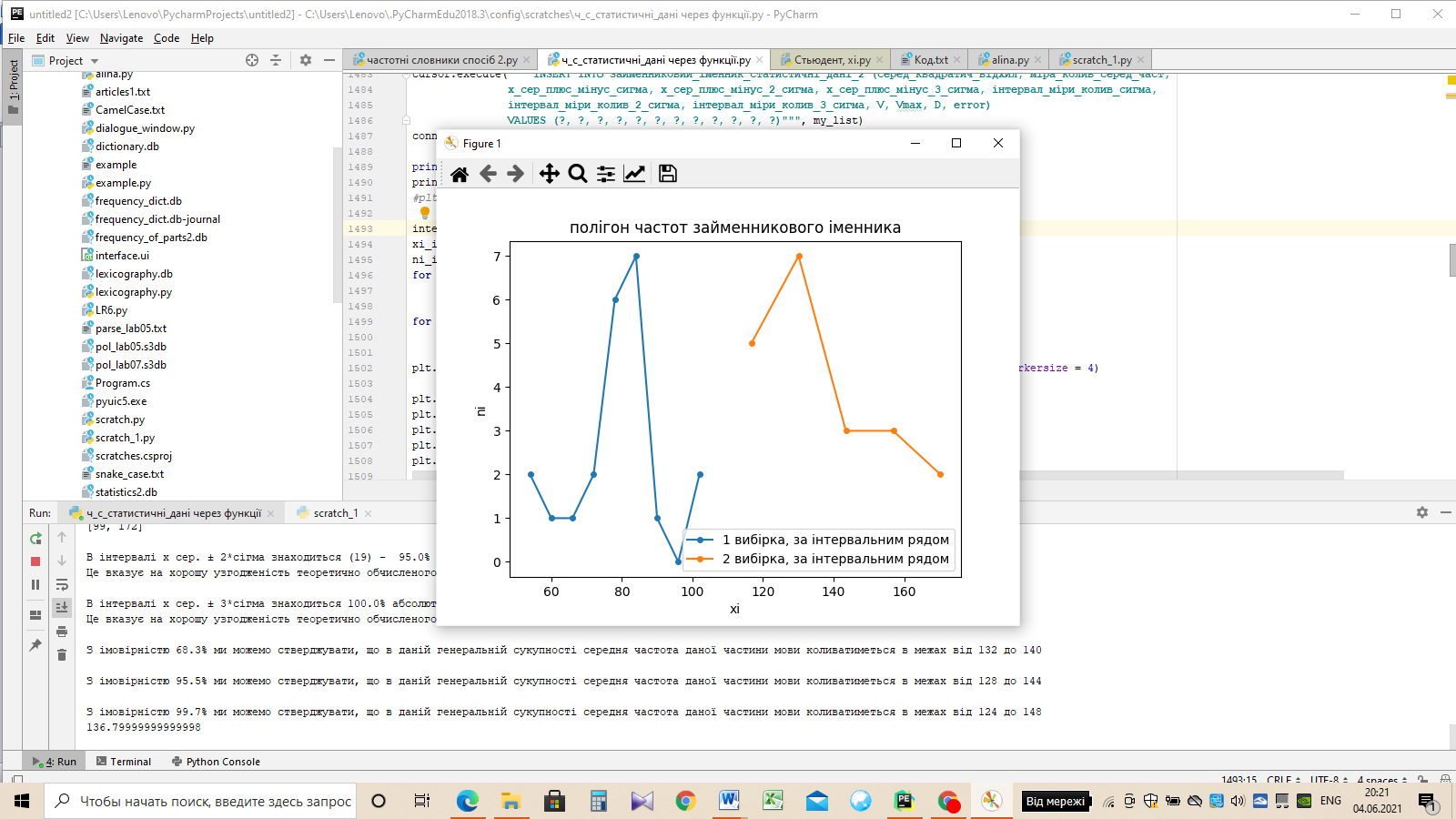
В другій вибірці найбільше абсолютних частот знаходиться в інтервалі з центром 180 (6), а також багато абсолютних частот припадає на інтервал з центрм 160(4). Беручи до уваги цей факт і те, що різниця міє мінімальним і максимальним значенням абсолютної частоти не значна, можна зробити висновок, що в другій вибірці дієслово розподілено рівномірно.

### Займенник

Варіаційний ряд займенника:

Полігон частот займенника

Абсолютні частоти займенника в першій вибірці коливаються в менших межах (51-105), ніж в другій (110-177) і не перетинаються зі значеннями абсолютних частот в другій вибірці. Майже всі значення в обох вибірках зустрічаються по разу, деякі – по два. Займенник також набагато частіше зустрічається в другій вибірці, ніж в першій.

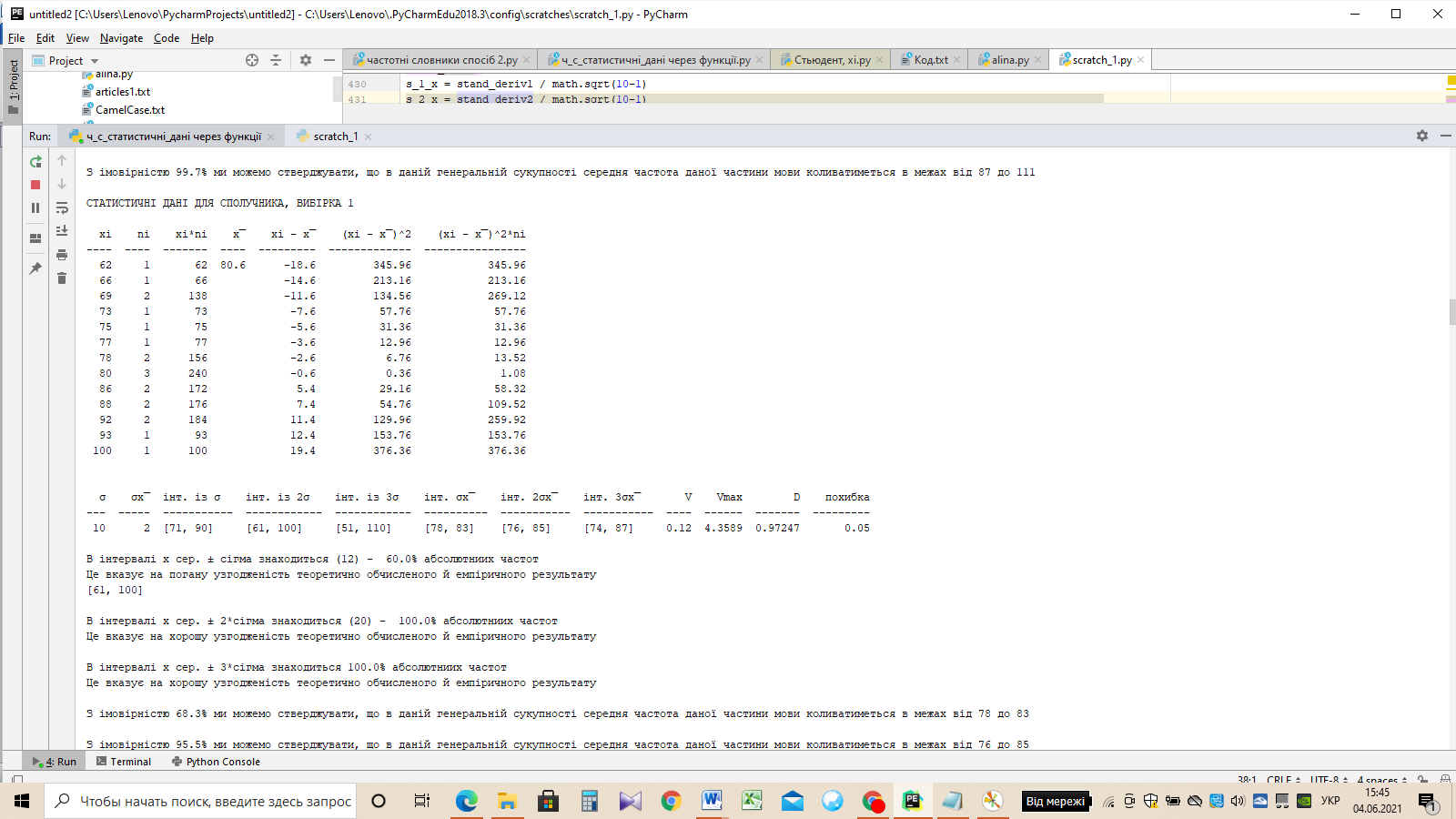
Але, щоб побачити особливості розподілу абсолютних частот у вибірка, необхідно побудувати полігони частот за інтервальними варіаційними рядами:

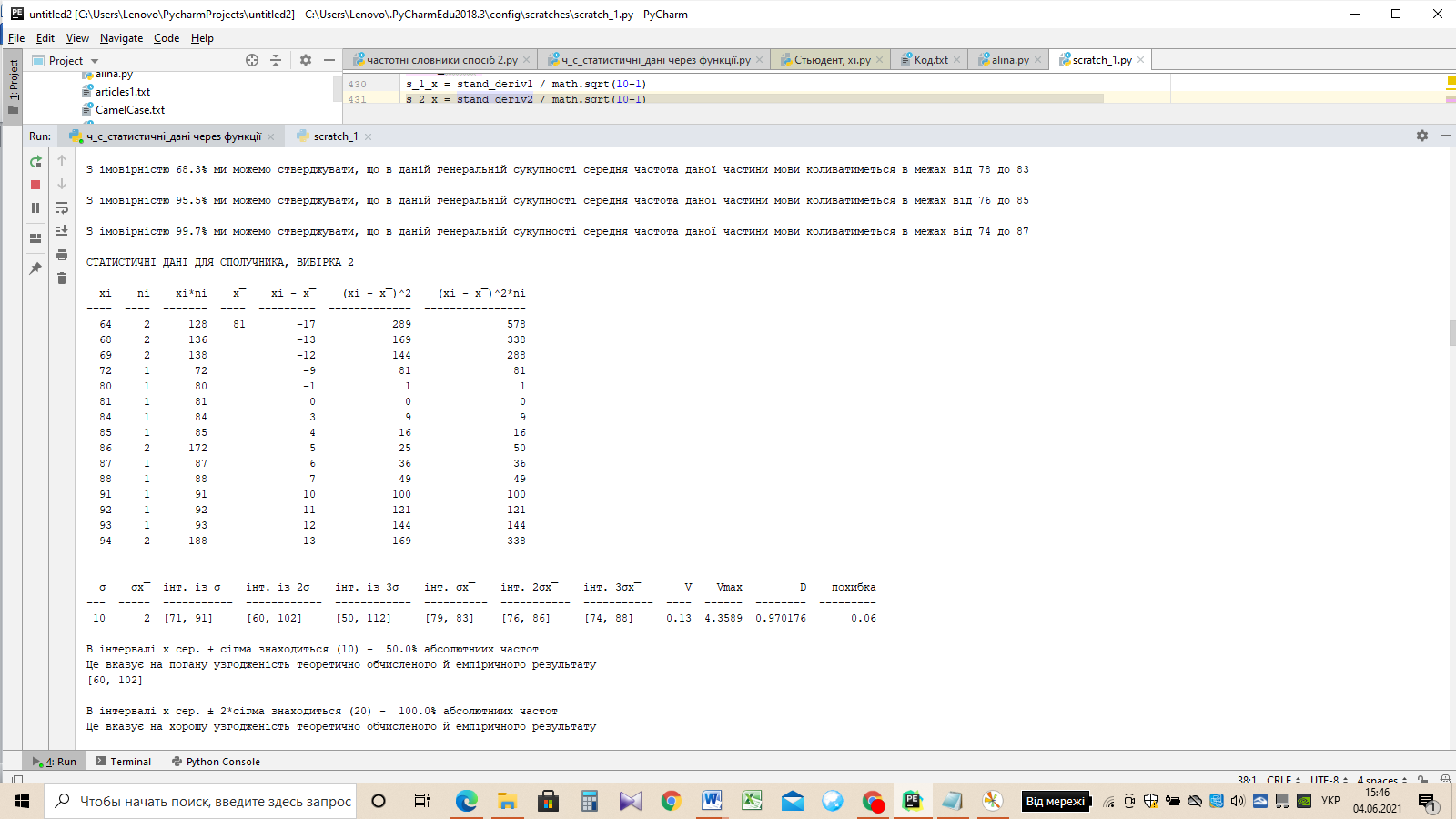
Тепер абсолютні частоти згруповані навколо одного центру, і легше зробити висновок про їх розподіл. В першій вибірці найбільше абсолютних частот знаходиться в інтервалі з центром 82 (7). На інші проміжки припадає не більше двох значень. Беручи до уваги цей факт і те, що різниця міє мінімальним і максимальним значенням абсолютної частоти не велика, можна зробити висновок, що в першій вибірці займенниик розподілений доволі рівномірно.

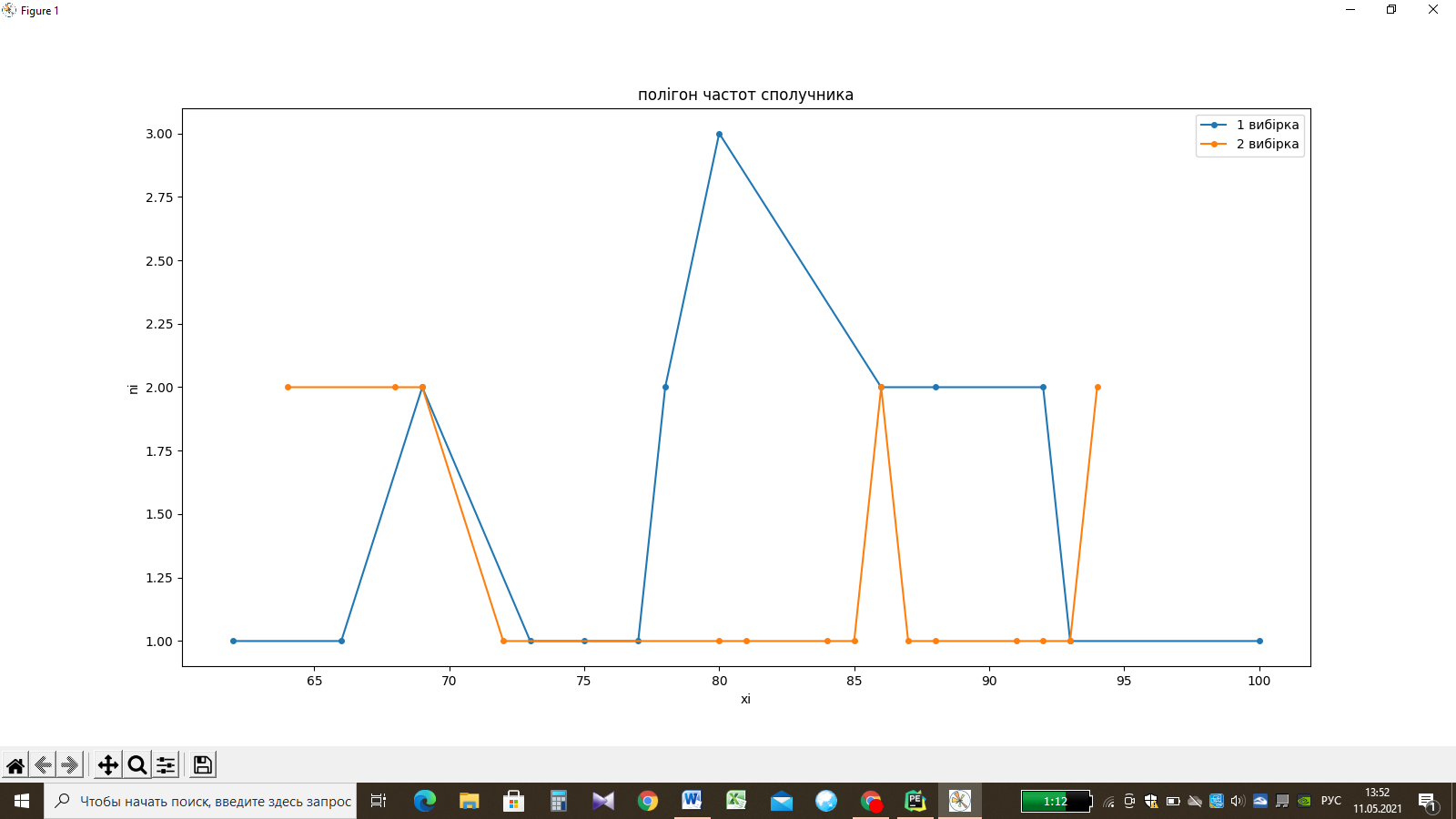
В другій вибірці найбільше абсолютних частот знаходиться в інтервалі з центром 130 (7. Беручи до уваги цей факт і те, що різниця міє мінімальним і максимальним значенням абсолютної частоти не значна, можна зробити висновок, що в другій вибірці займенниик розподілений рівномірно.

### Сполучник

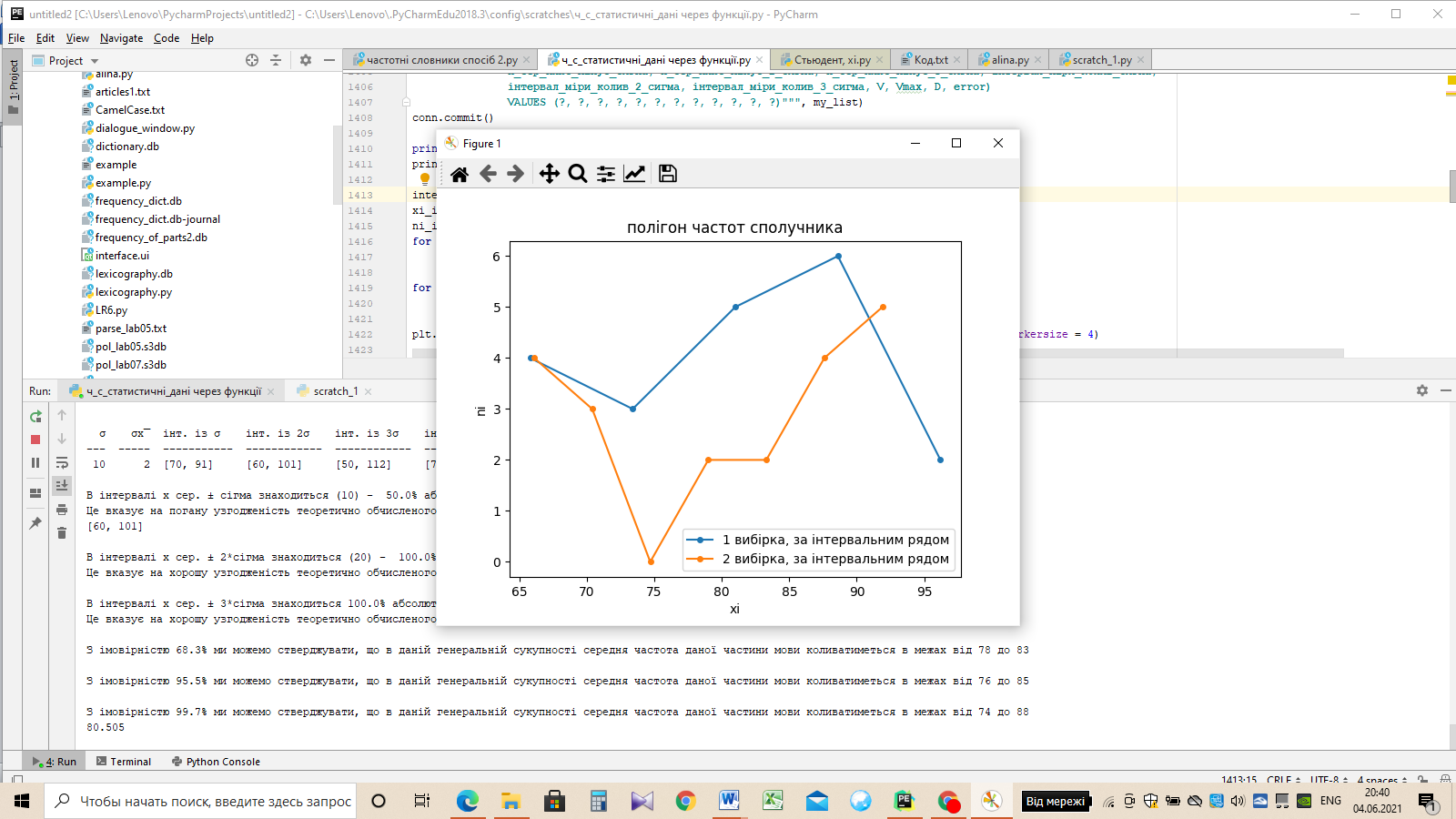
Варіаційний ряд сполучника:





Полігон частот сполучника:

Межі коливання абсолютних частот сполучника у двох вибірках перетинаються і дуже схожі. В першій вони знаходяться в межах 62-100, а в другій – 64-94. Середнє значення частоти для цієї частини мови в двох вибірках приблизно однакове. Отже, сполучник однаково зустрічається в них.

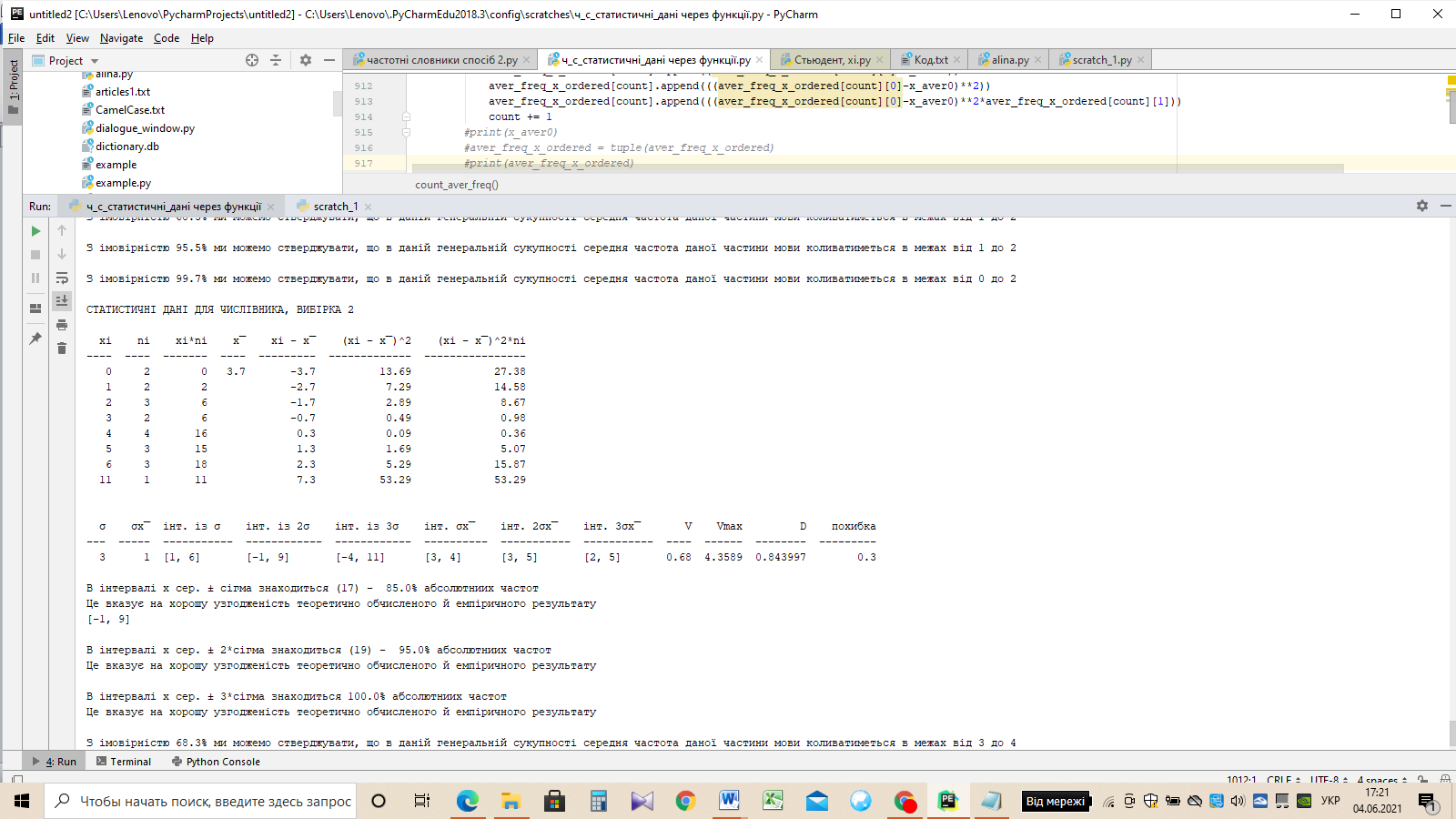
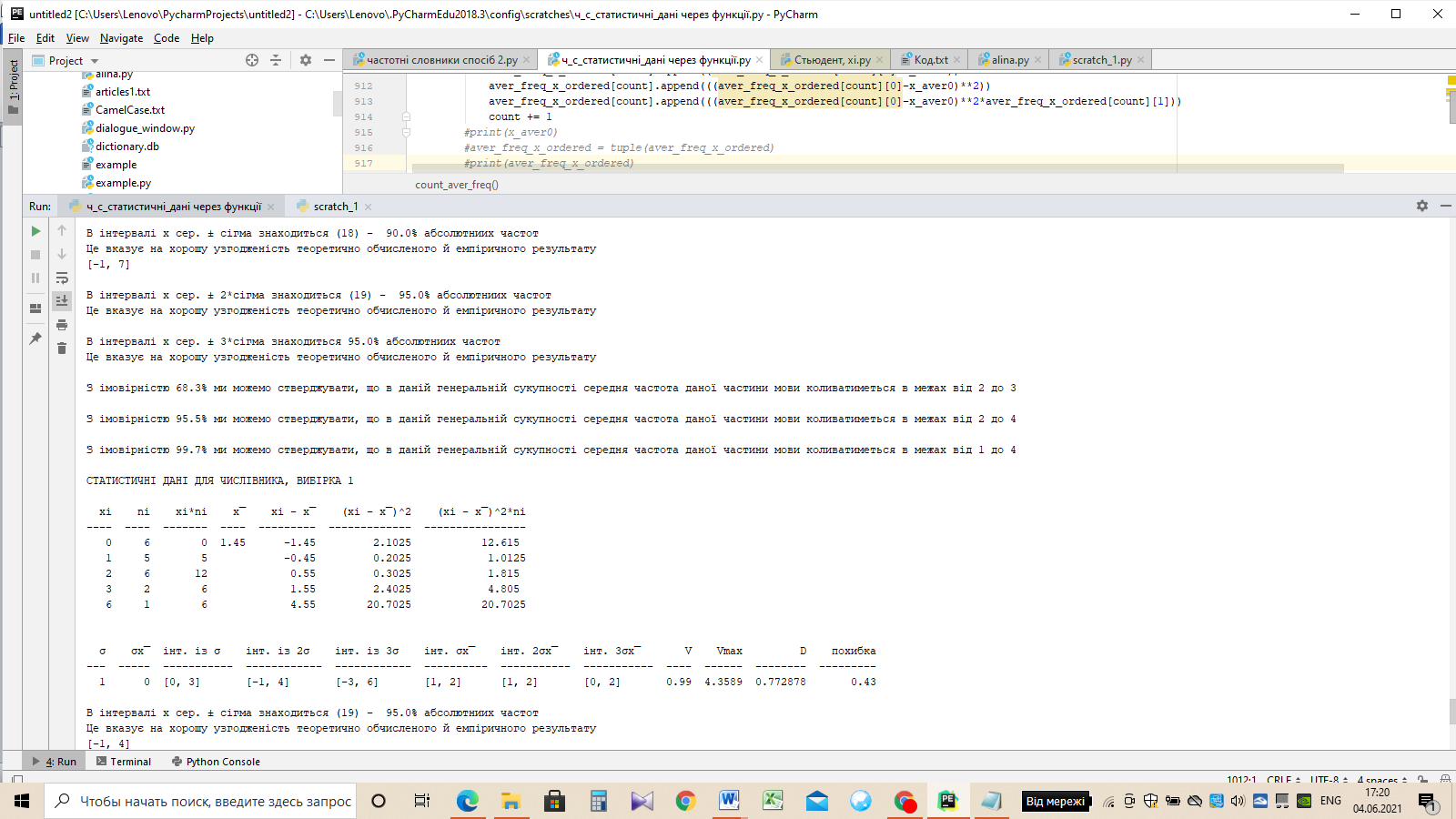
Щоб побачити особливості розподілу абсолютних частот, побудуймо полігони частот за інтервальним варіаційним рядом:

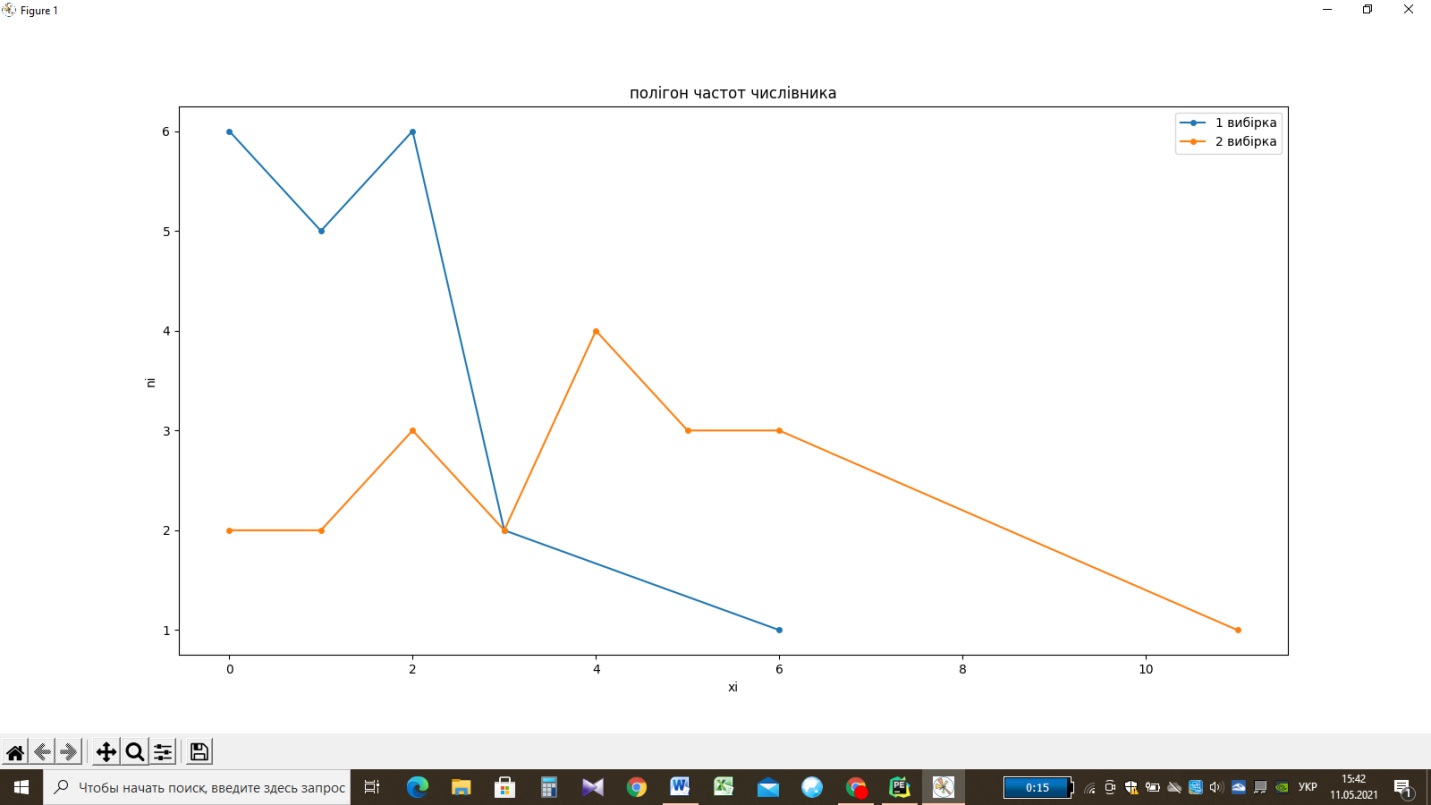
Тепер абсолютні частоти згруповані навколо одного центру, і легше зробити висновок про їх розподіл. В першій вибірці найбільше абсолютних частот знаходиться в інтервалі з центром 88 (6). На інтервал з серединою 80 припадає 5 значень абсолютних частот. Беручи до уваги цей факт і те, що різниця міє мінімальним і максимальним значенням абсолютної частоти не велика, можна зробити висновок, що в першій вибірці сполучник розподілений доволі рівномірно.

В другій вибірці сполучник розподілений менш рівномірно. Найбільше абсолютних частот знаходиться в інтервалі з центром 93 (5). На інтервали з серединою 82 і 62 припадає по 4 значення абсолютних частот.

Варто зазначити, що, хоч середня частота сполучника в вибірках співпадає, розподілена ця частина мови в них по-різному.

### Числівник

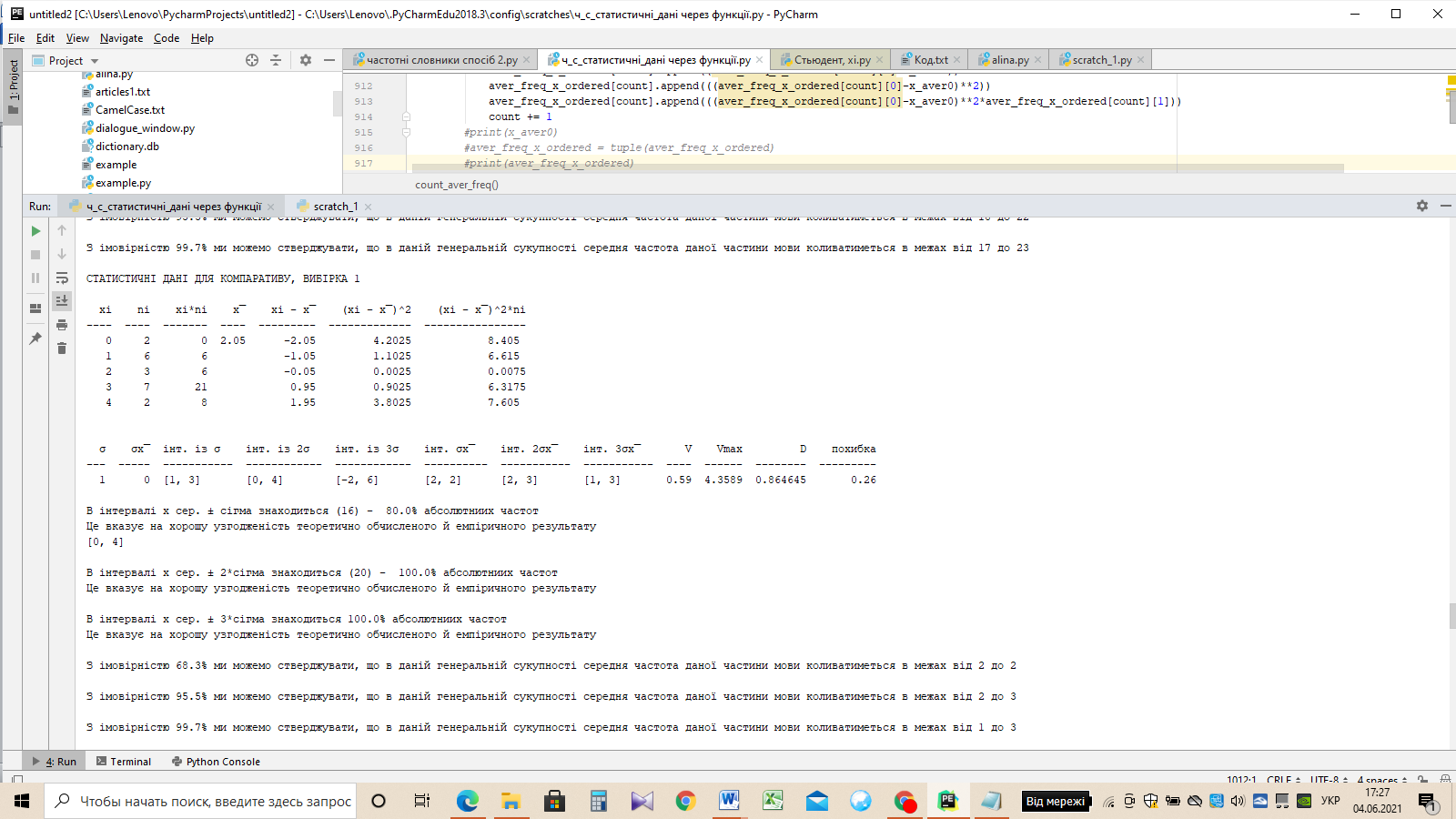
Варіаційний ряд числівника:

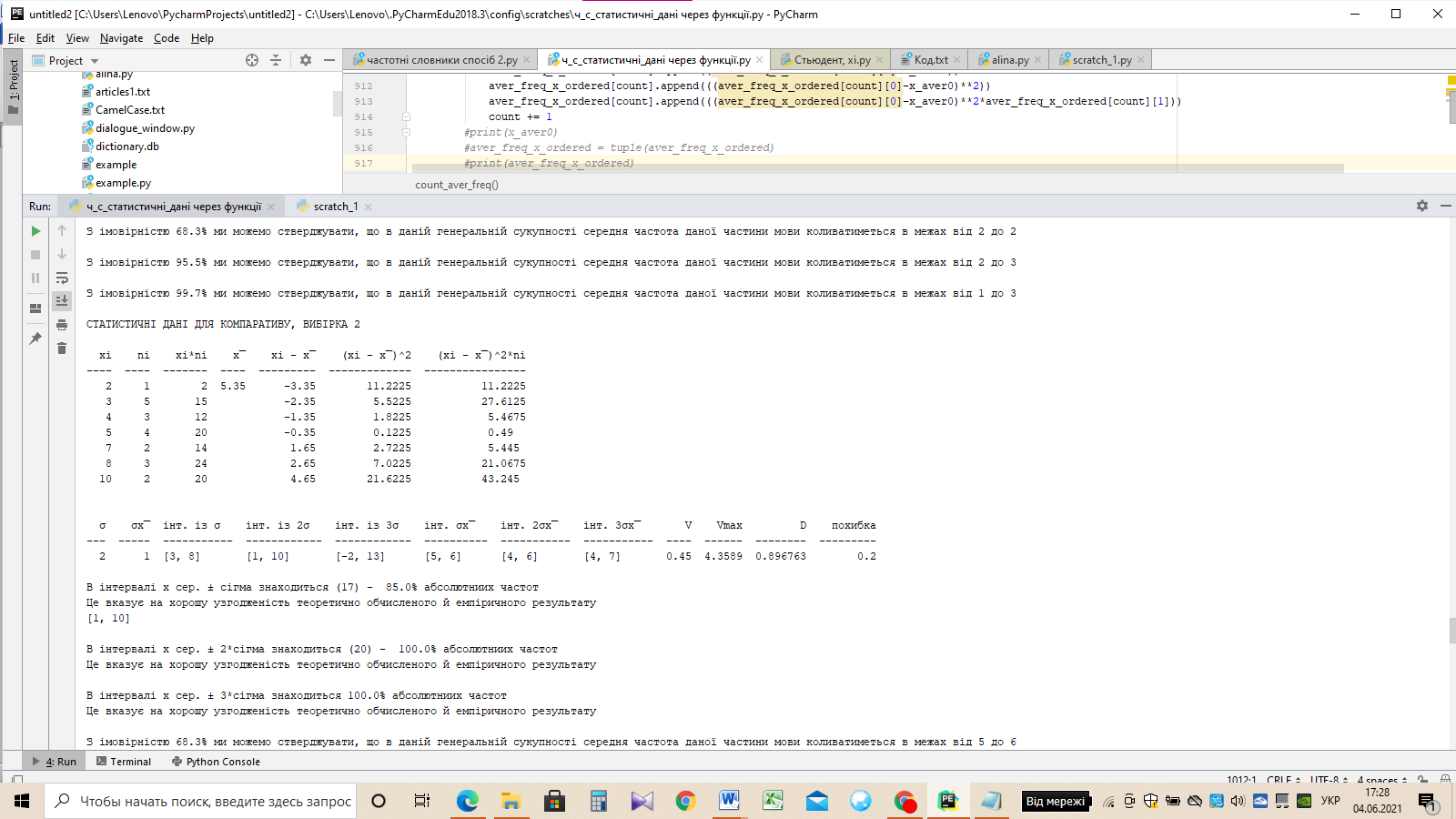
Полігон частот числівника:

Абсолютні частоти числівника в першій вибірці коливаються в менших межах (0-6), ніж в другій (0-11) , але перетинаються зі значеннями абсолютних частот в другій вибірці. Найчастіше в першій вибірці зустрічається від 0 до 2 числівників на підвибірку, а в другій – 4.Беручи до уваги цей факт і те, що різниця міє мінімальним і максимальним значенням абсолютної частоти маленька, можна зробити висновок, що в обох вибірках числівник розподілений доволі рівномірно.

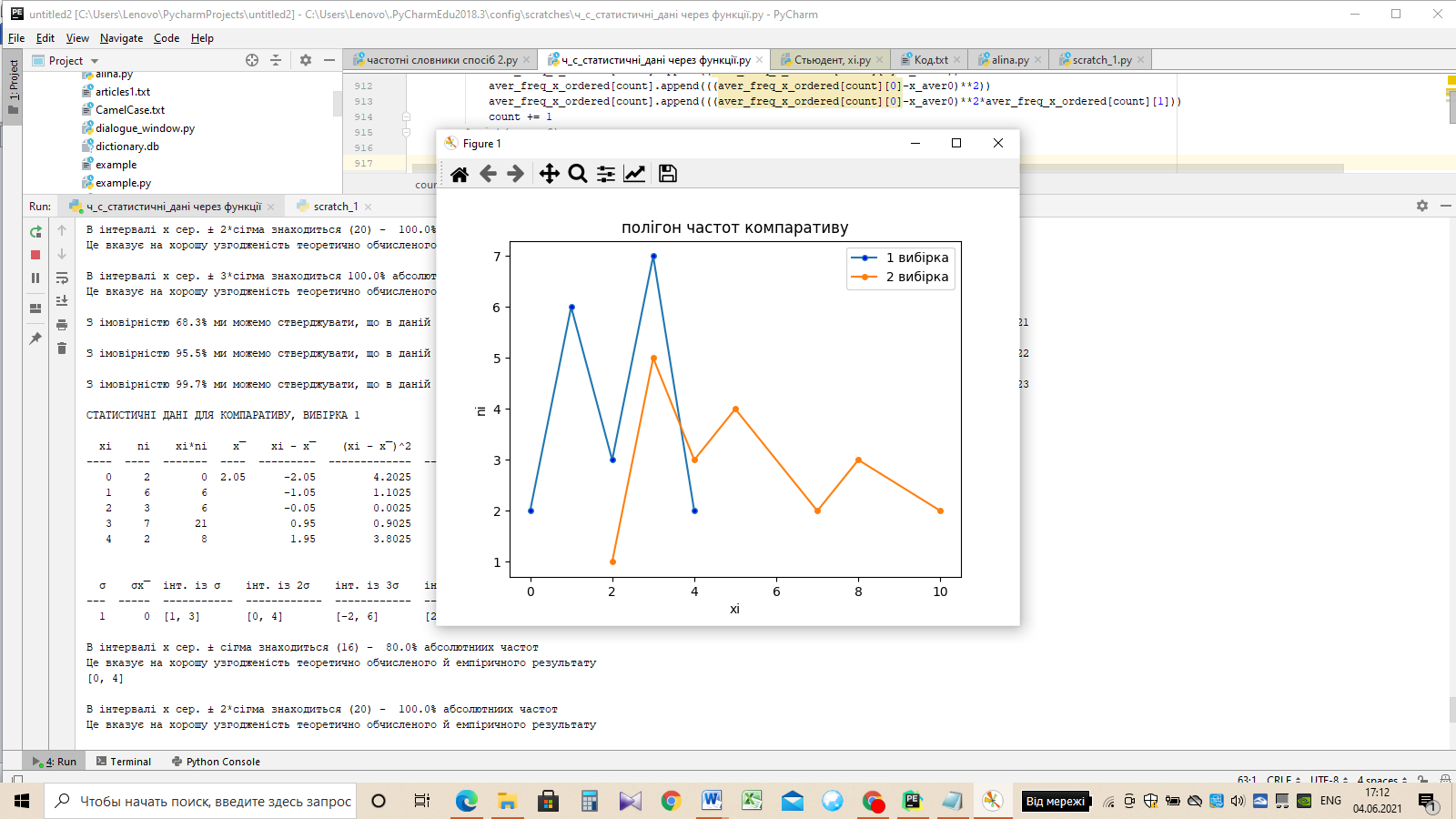
### Компаратив

Варіаційний ряд компаративу:



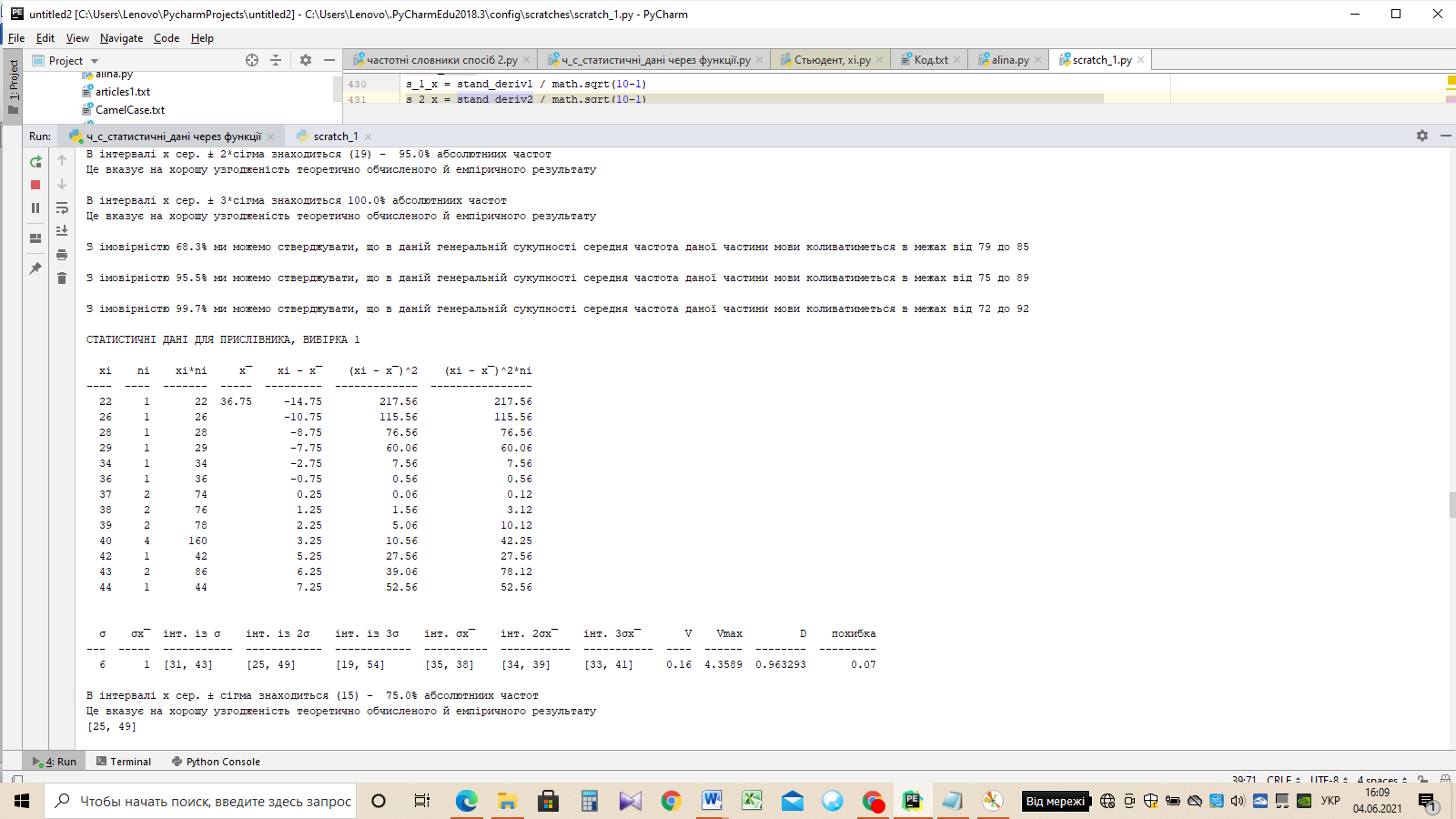


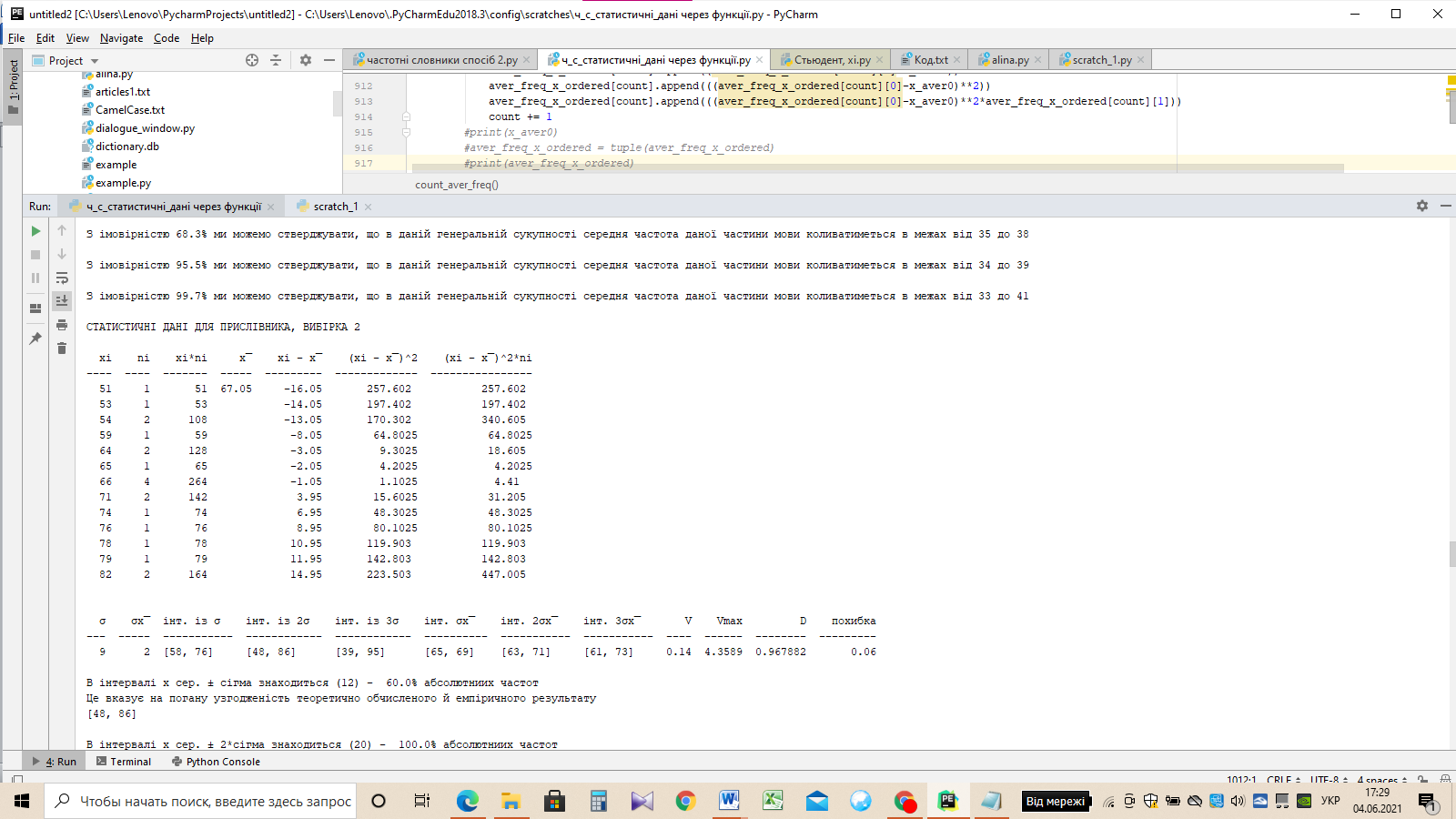
Полігон частот компаративу:



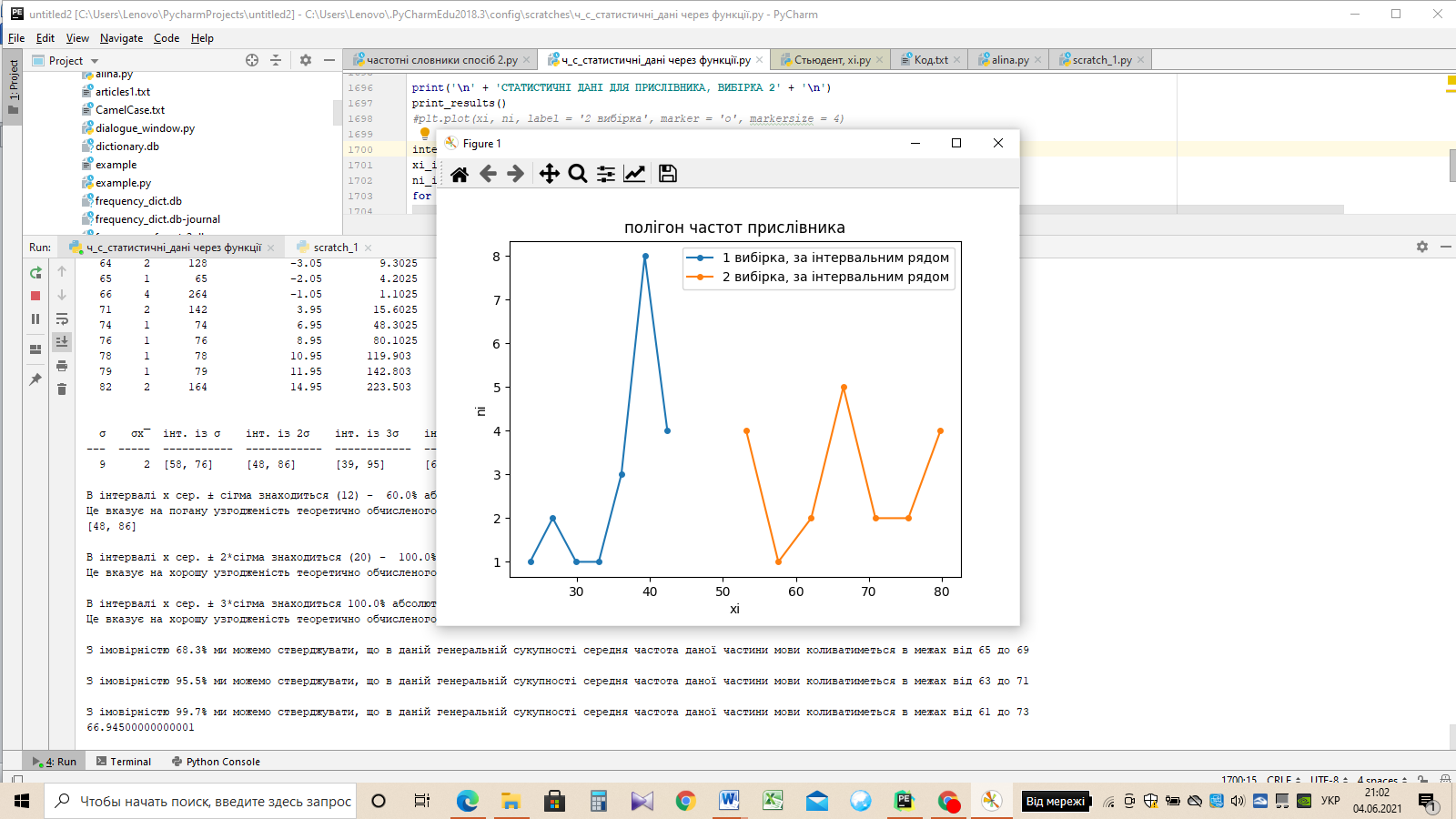
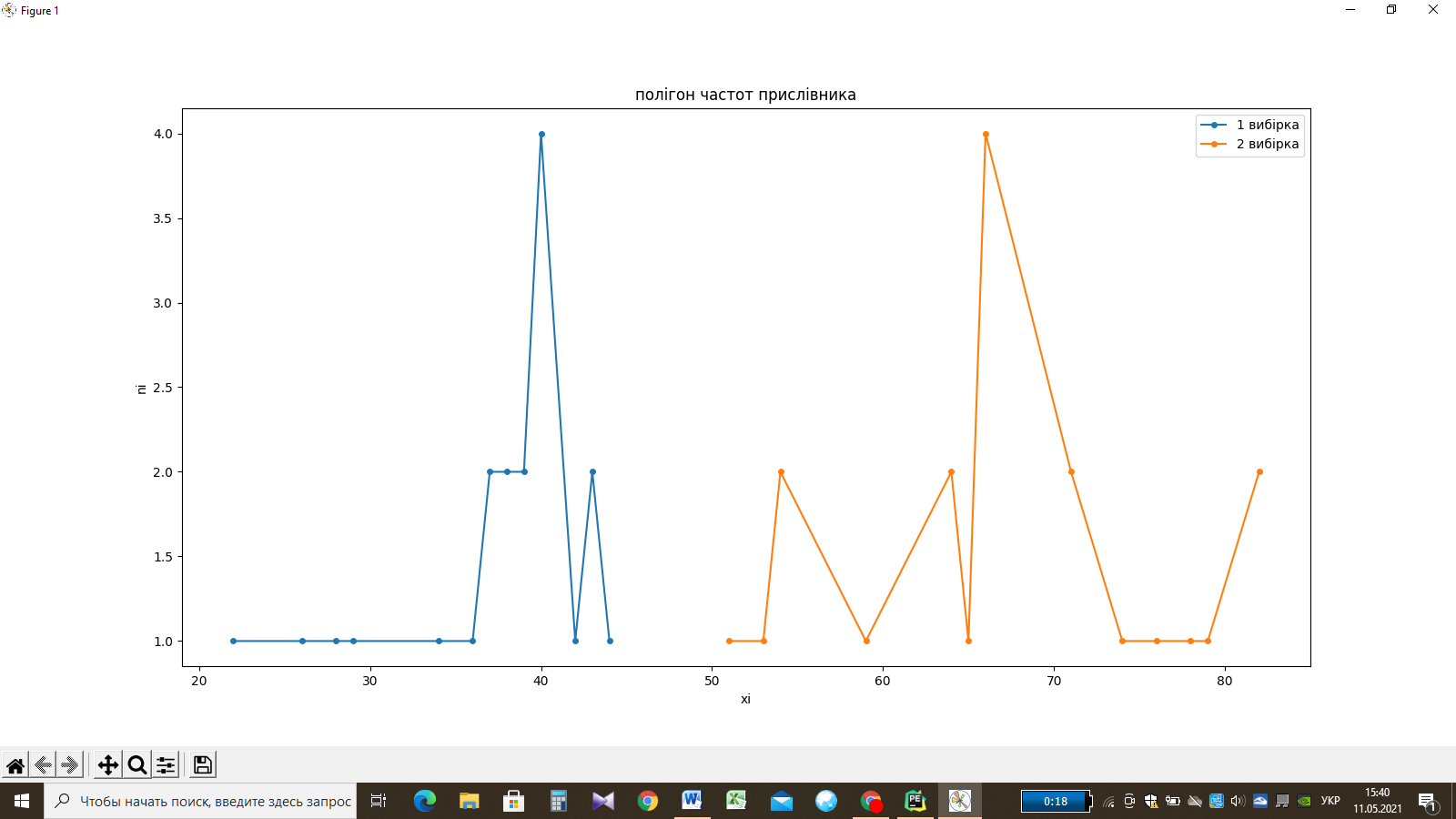
Значення абсолютних частот компаративу в першій вибірці знаходяться в межах від 0 до 4. Значення 3 зустрічається найбільше разів (7), також часто в підвибірках зустрілося по 1 компаративу (6 разів). Отже, можна зробити висновок, що компаратив рівномірно розподілений по підвибіркам. У другій вибірці його абсолютні частоти обмежені значеннями 2 і 10. Найбільше абсолютних частот, що дорівнюють 3 (їх 5); чотири абсолютні частоти зі значенням 5 і три зі значенням 8. У другій вибірці ця частина мови зустрічається частіше. Отже, можна зробити висновок, що в першій вибірці компаратив розподілений доволі рівномірно, а от в другій – ні.

### Прислівник

Варіаційний ряд прислівника:



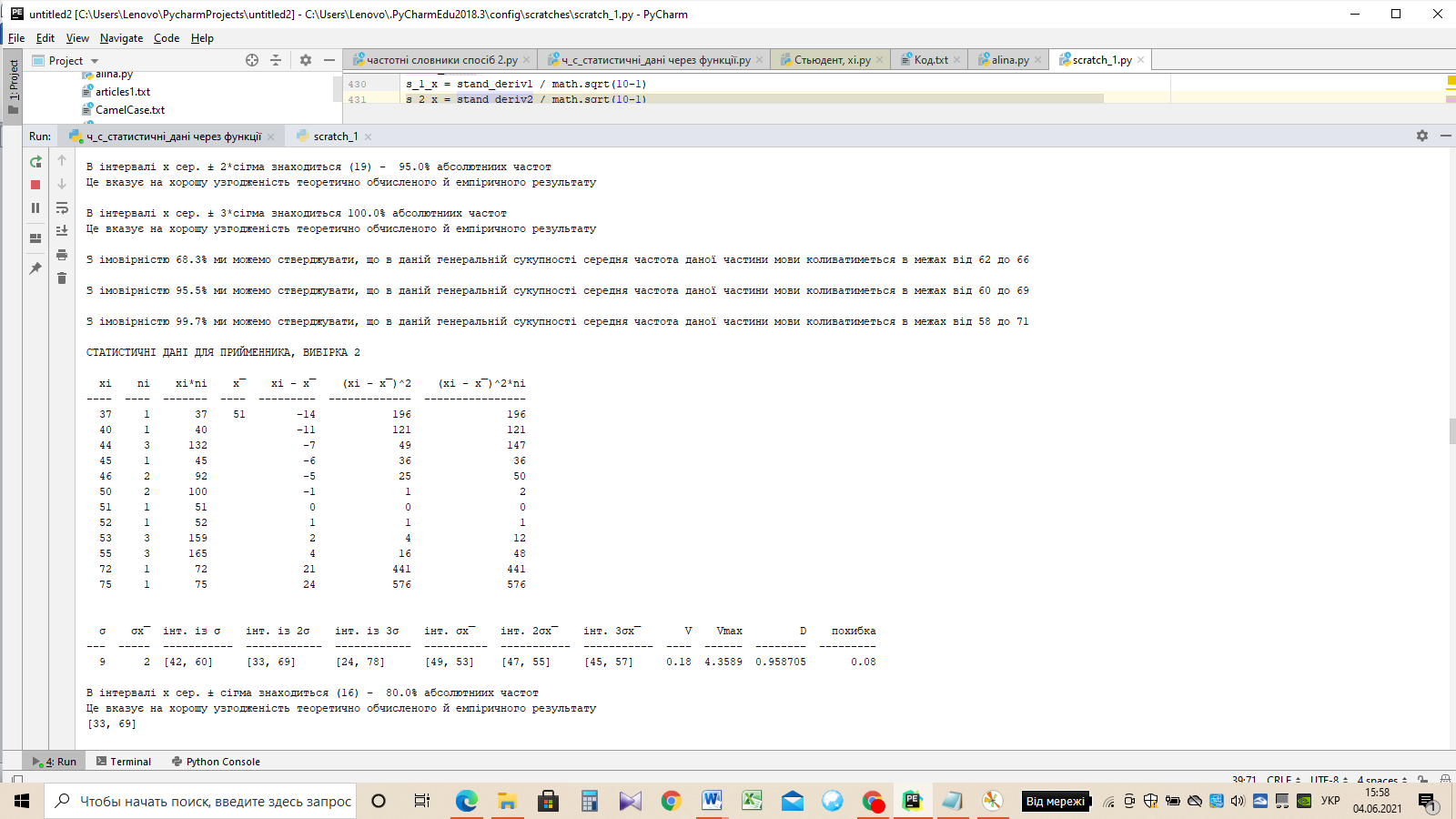
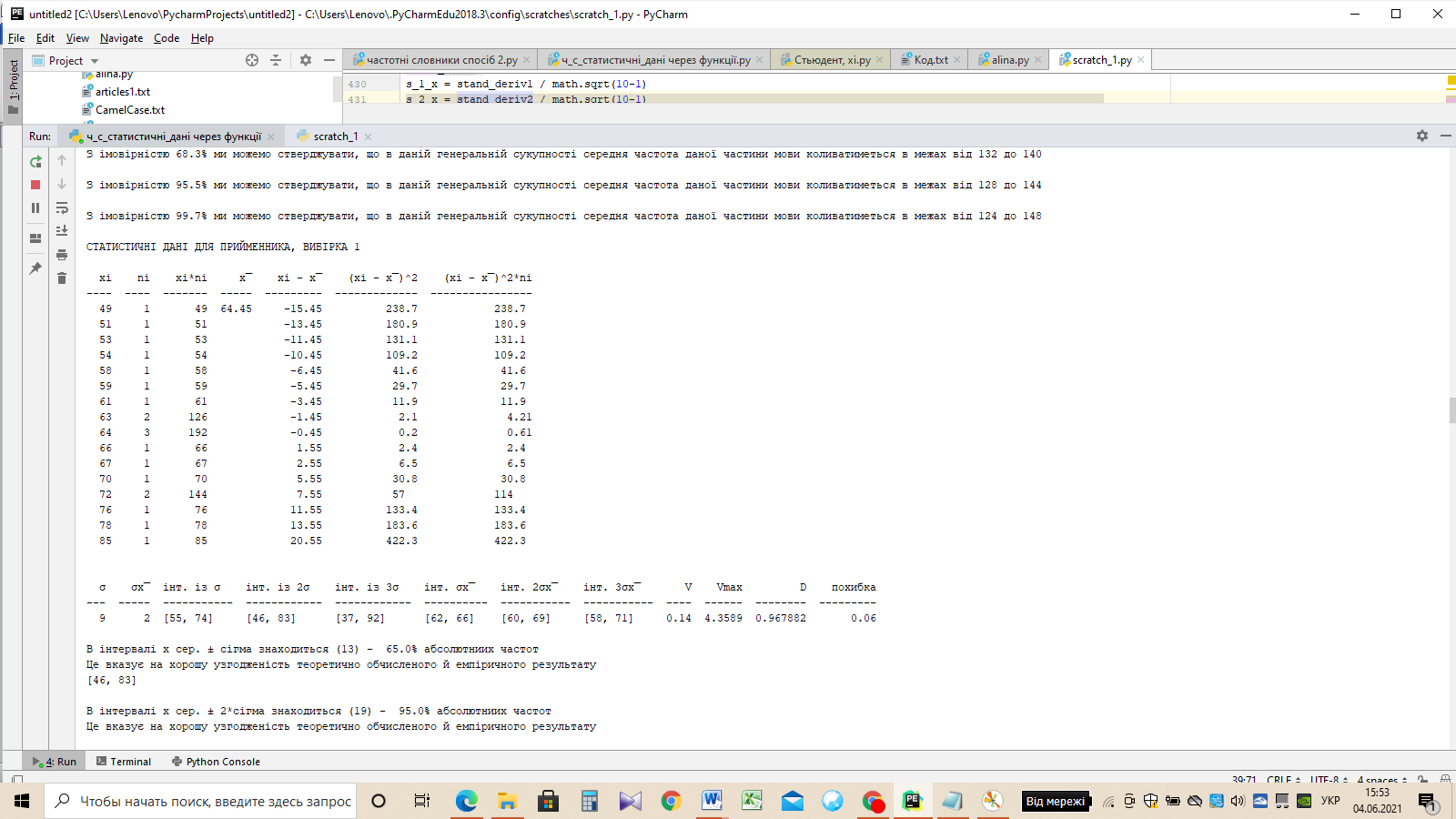
Полігон частот прислівника:

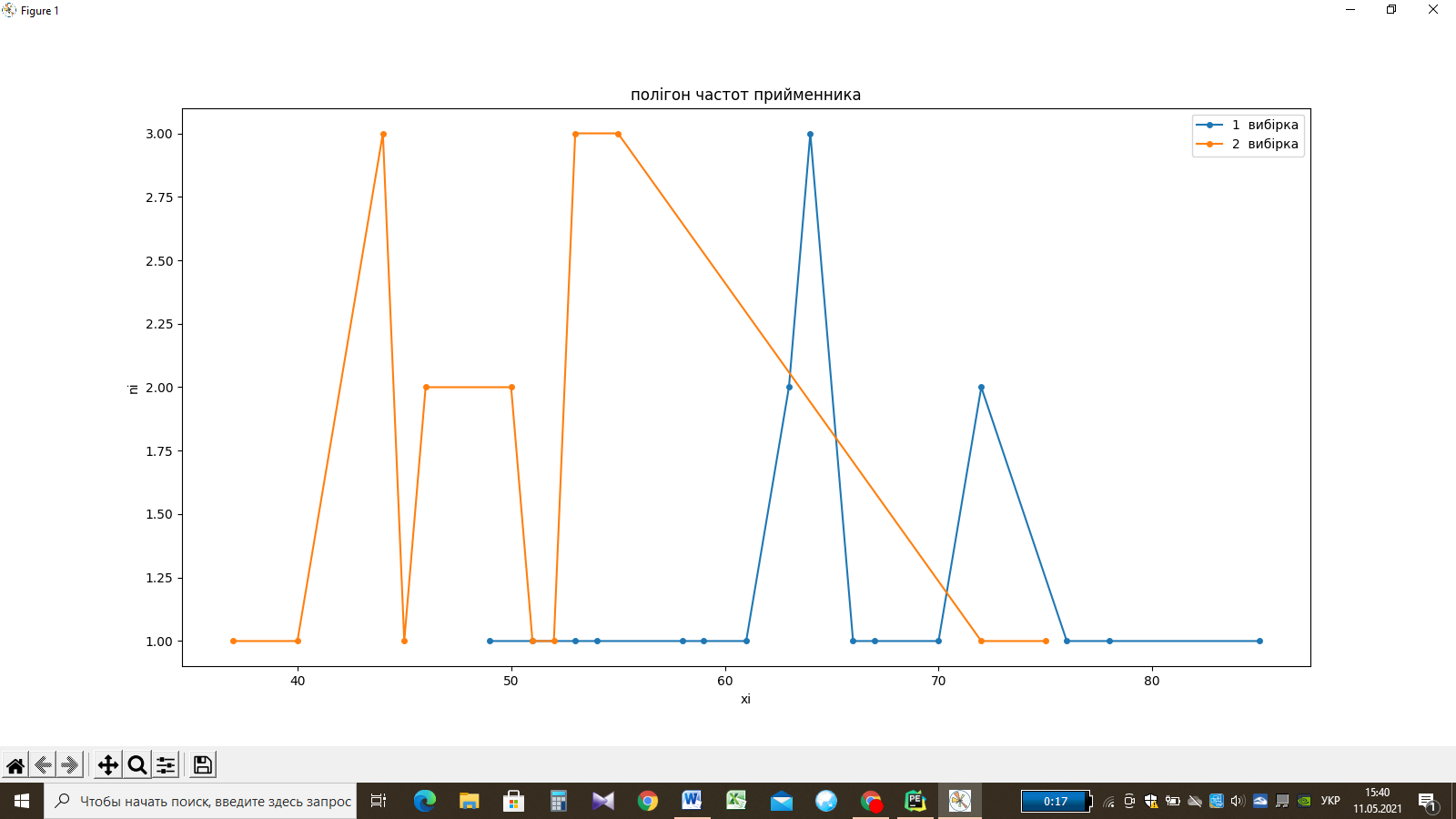
Значення абсолютних частот прислівника в першій вибірці знаходяться в межах від 22 до 44, а в другій – від 51 до 82. Частіше ця частина мови зустрічається в другій вибірці.

Побудуймо полігони частот за інтервальним варіаційним рядом:

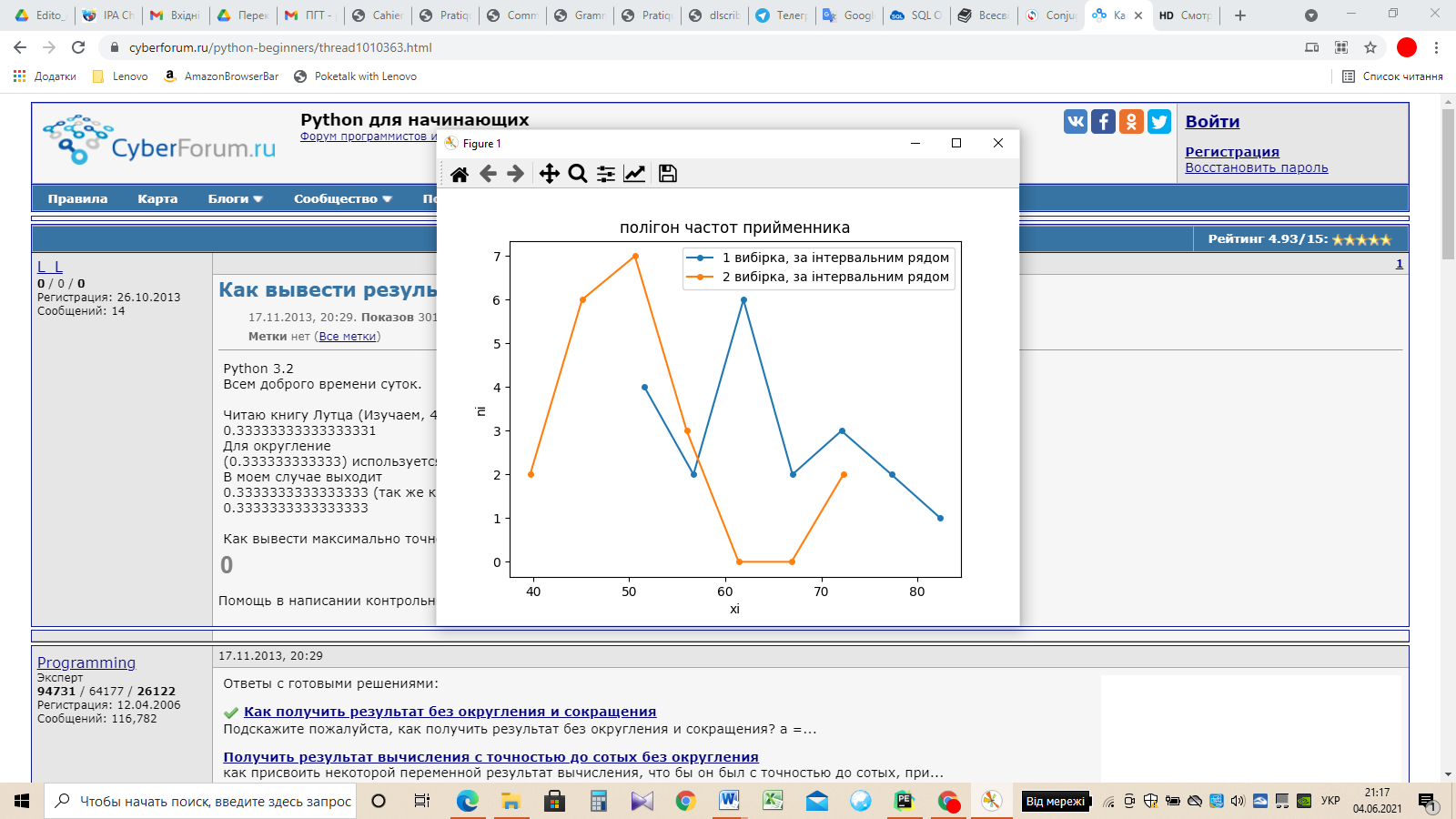
На графіку видно, що в першій вибірці в інтервалі з серединою 40 кількість абсолютних частот різко зростає до 8. Оскільки мінімальне та максимальне значення їх відрізняється не сильно, можна зробити висновок про рівномірний розподіл прислівника в цій вибірці. В другій вибірці найбільше абсолютних частот в інтервалах з серединою 68 (5), 53 (4) та 80 (4), що свідчить про менш рівномірний розподіл прислівника в даній вибірці.

### Прийменник

Варіаційний ряд прийменника:

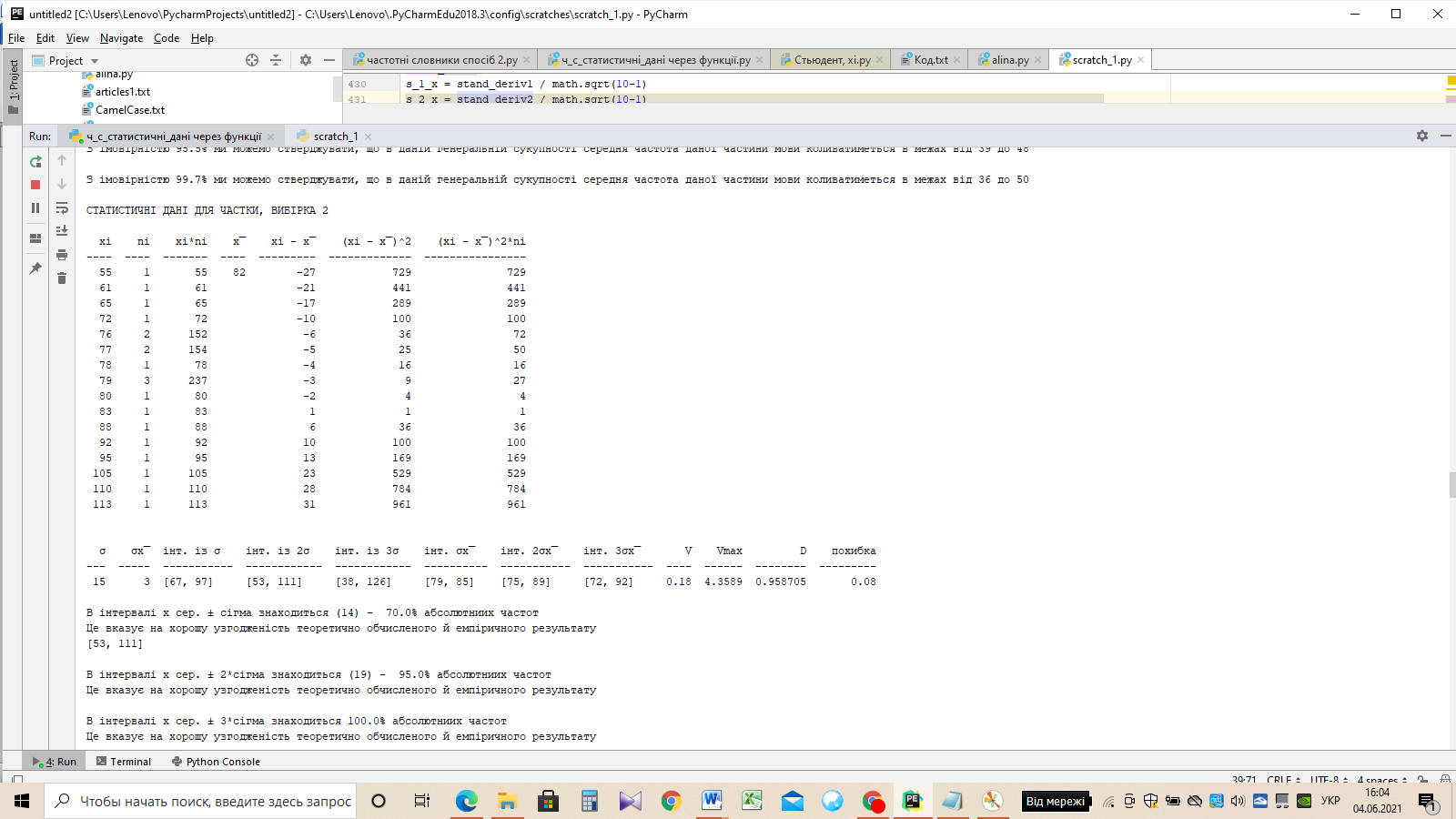
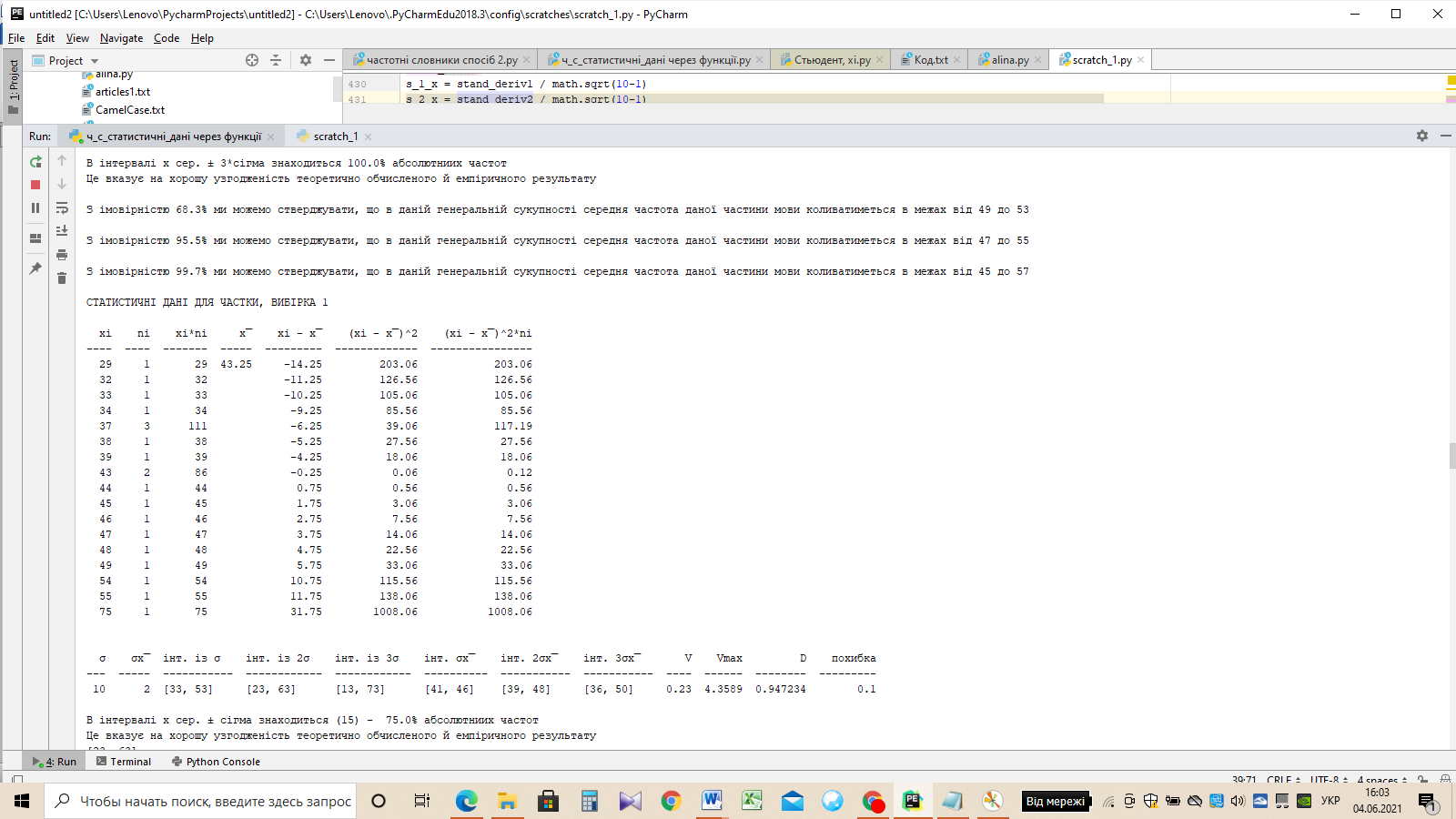
Полігон частот прийменника

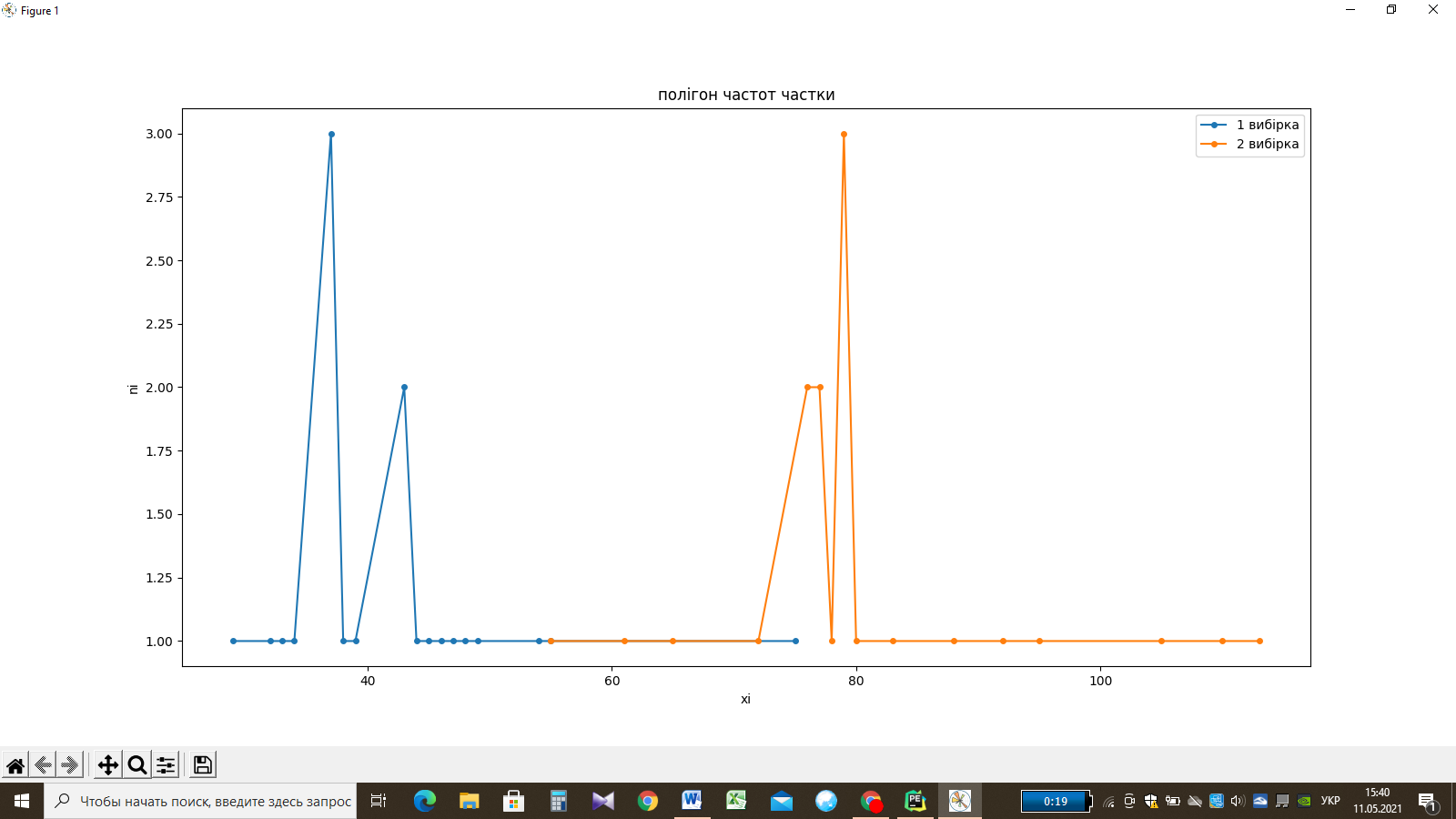
Значення абсолютних частот прийменника в першій вибірці знаходяться в межах від 49 до 85, а в другій – від 37 до 75, ці значення перетинаються. Частіше ця частина мови зустрічається в першій вибірці.

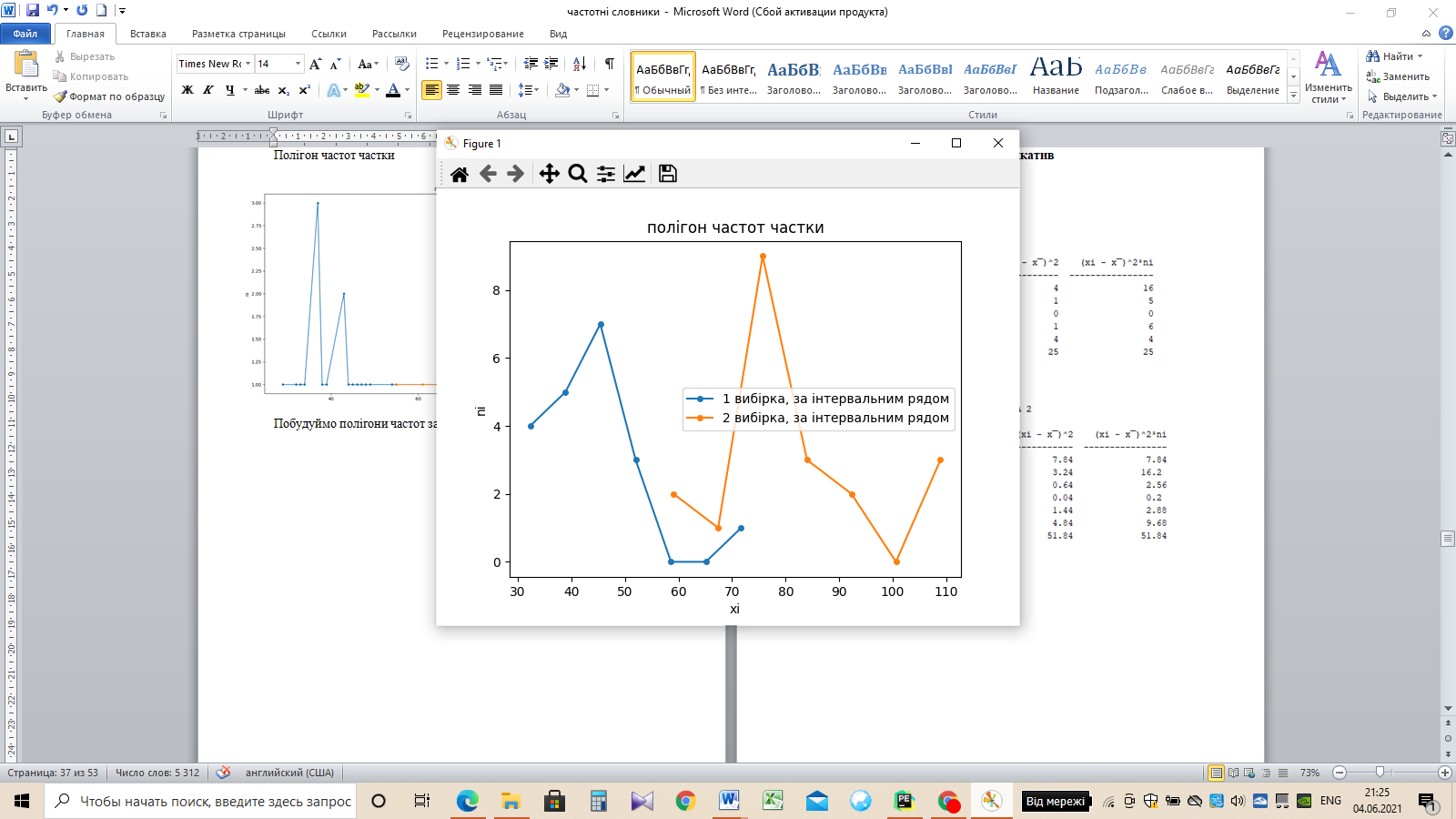
Оскільки на даному графіку розподіл частот не рівномірний, побудуймо полігони частот за інтервальними варіаційними рядами:

На графіку видно, що найбільше абсолютних частот в першій вибірці в інтервалі з серединою 65 (5), що свідчить про менш рівномірний розподіл прийменника в даній вибірці. . В другій вибірці найбільше абсолютних частот в інтервалі з серединою 55 кількість абсолютних частот різко зростає до 7. Оскільки мінімальне та максимальне значення їх відрізняється не сильно, можна зробити висновок про рівномірний розподіл прийменник в цій вибірці.

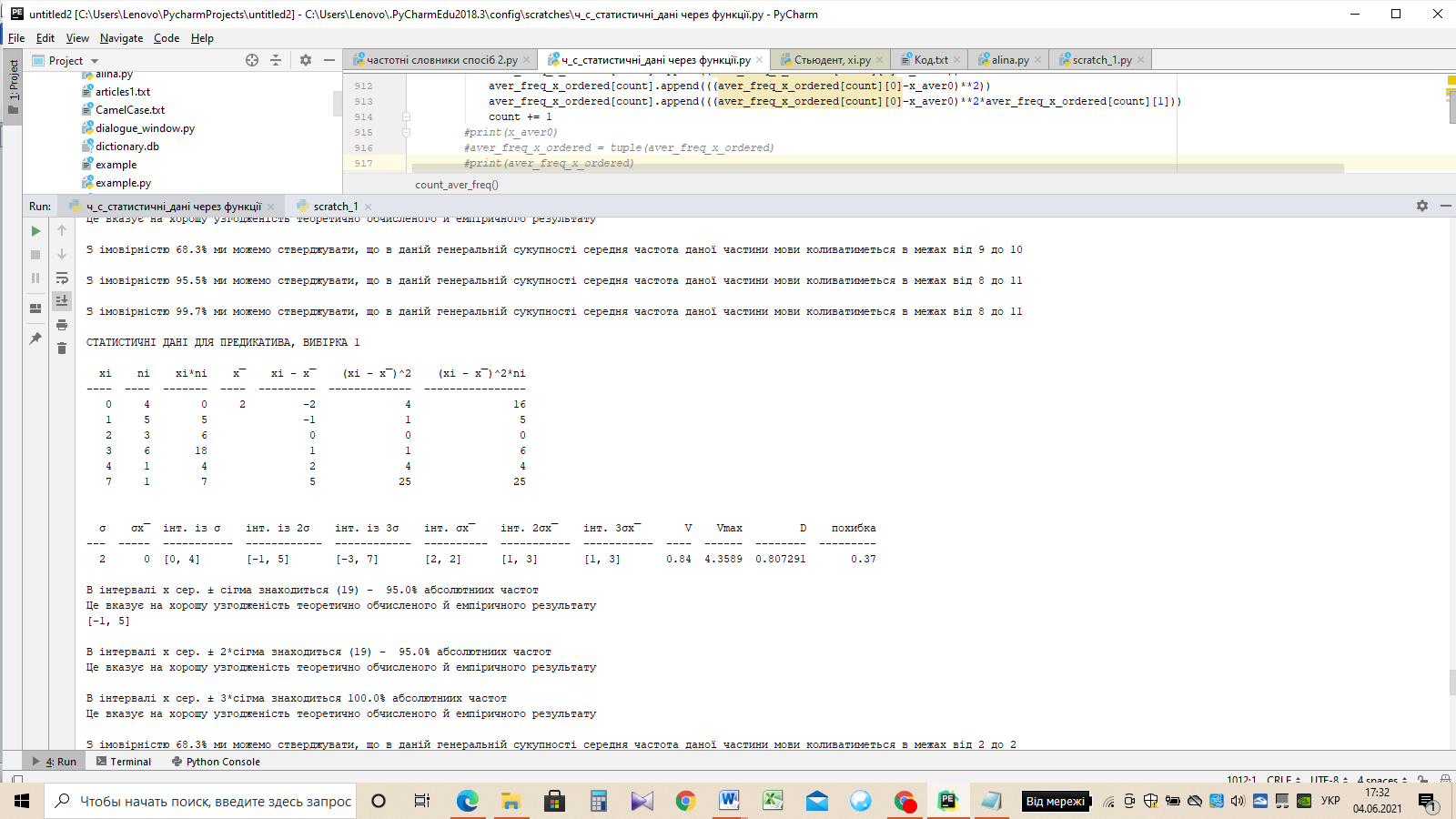
### Частка

Варіаційний ряд частки:

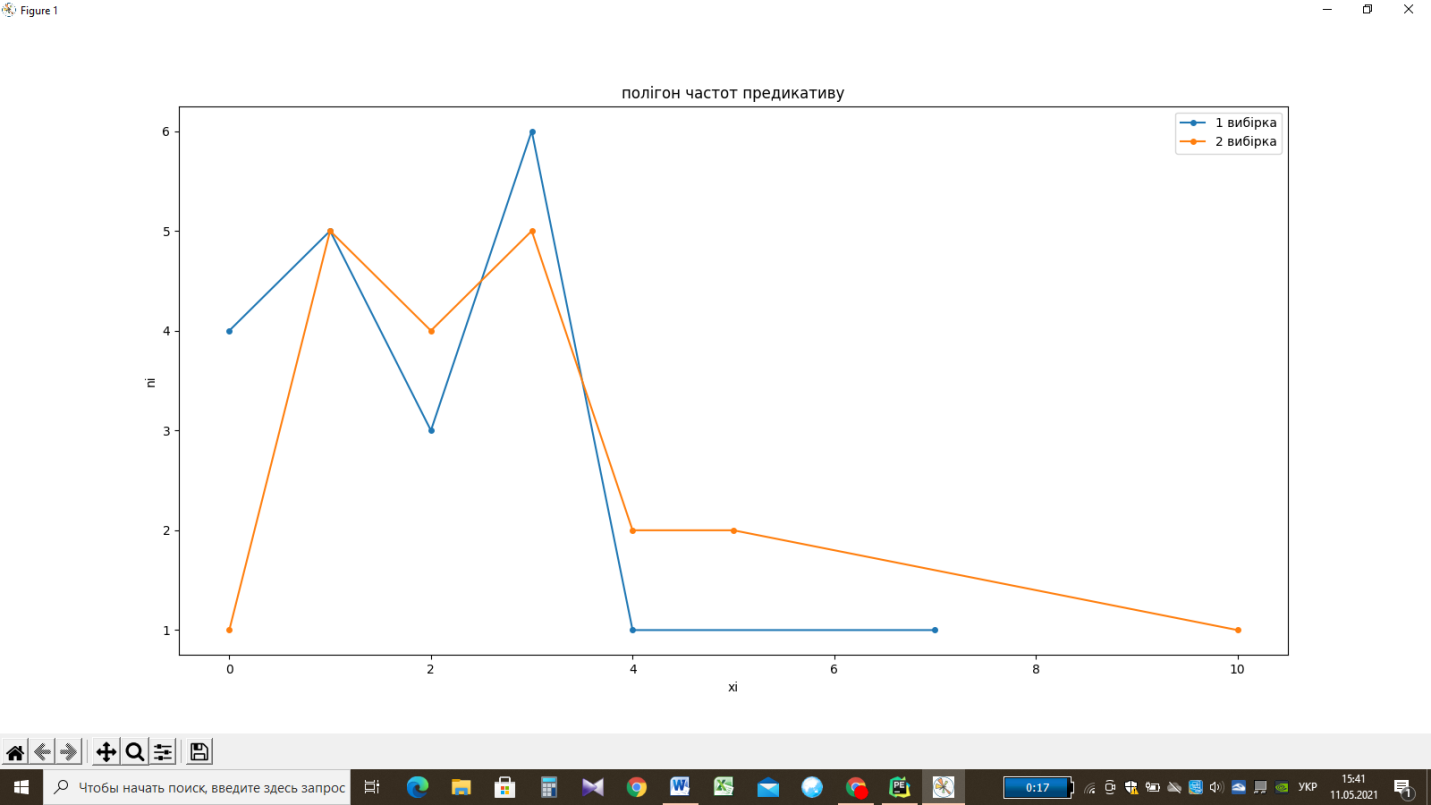
Полігон частот частки

Побудуймо полігони частот за інтервальними варіаційними рядами:

### Предикатив

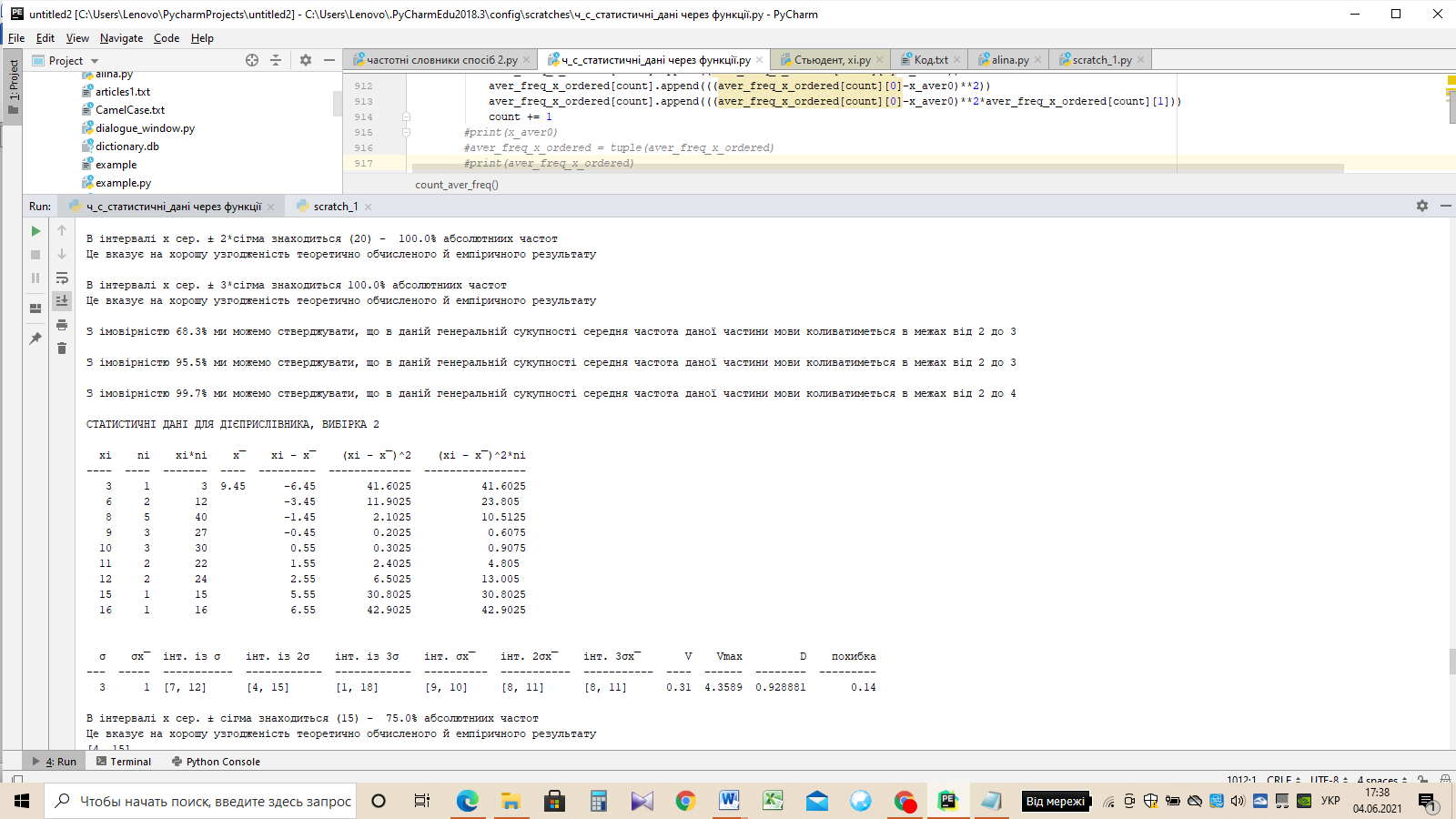
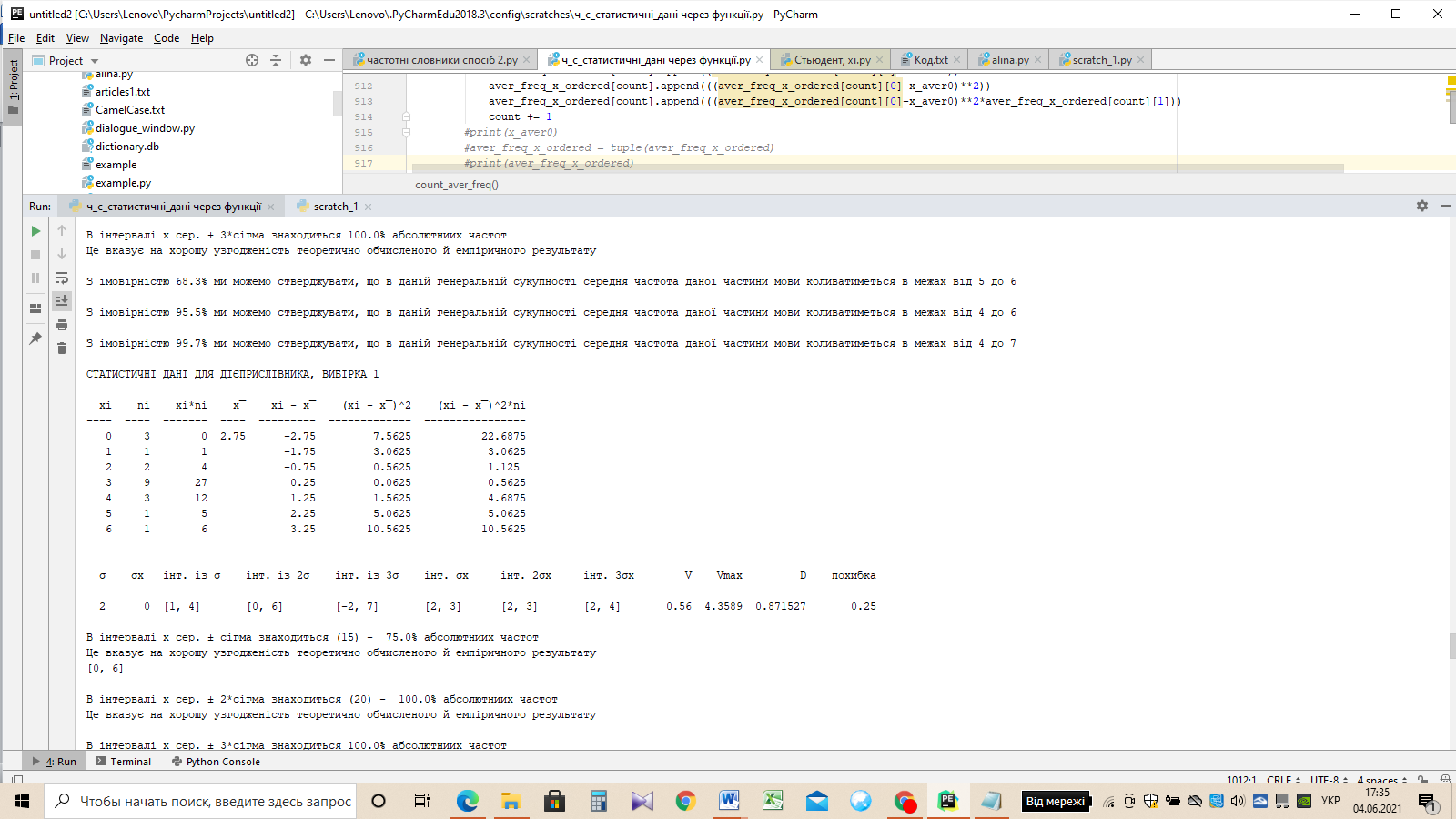
Варіаційний ряд предикатива:



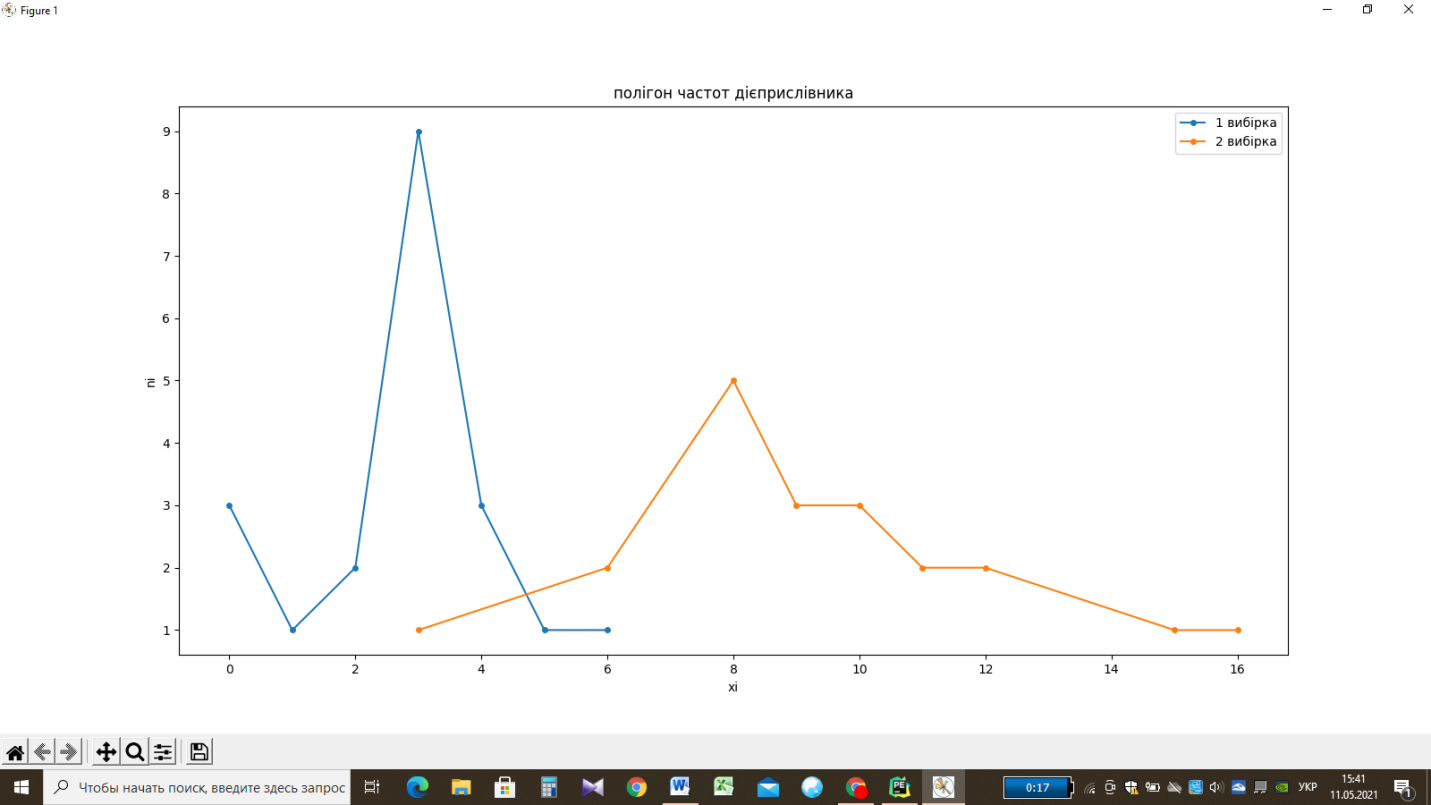
Полігон частот предикативу

### Дієприслівник

Варіаційний ряд дієприслівника:



Полігон частот дієприслівника



# РОЗДІЛ 2. Обчислення основних статистичних характеристик частин мови

У цій частині роботи головною задачею є створити програмний код, який обчислює основні статистичні характеристики частин мови в вибірках.

Спочатку було виконано з’єднання з базою даних, де знаходяться частотні словники.

Потім було створено потрібні таблиці. Вони знаходяться в цій самій БД. Інформація для кожної частини мови в вибірці записано в дві таблиці. Перша містить варіаційний ряд та деякі величині, необхідні для обчислення середнього квадратичного відхилення. Друга складається безпосередньо зі статистичний характеристик.

Таблиці було створено за таким програмним кодом:

cursor.execute(**'''CREATE TABLE IF NOT EXISTS іменник\_середня\_частота  
 (id INTEGER PRIMARY KEY NOT NULL,  
 xi INTEGER,  
 ni INTEGER,  
 xini INTEGER,  
 x\_сер INTEGER,  
 різниця\_xi\_та\_x\_сер INTEGER,  
 квадрат\_різниці\_xi\_та\_x\_сер INTEGER,  
 квадрат\_різниці\_xi\_та\_x\_серni INTEGER)  
'''**)  
conn.commit()  
  
cursor.execute(**'''CREATE TABLE IF NOT EXISTS іменник\_статистичні\_дані  
 (id INTEGER PRIMARY KEY NOT NULL,  
 серед\_квадратич\_відхил REAL,  
 міра\_колив\_серед\_част REAL,  
 x\_сер\_плюс\_мінус\_сигма TEXT,  
 x\_сер\_плюс\_мінус\_2\_сигма TEXT,  
 x\_сер\_плюс\_мінус\_3\_сигма TEXT,  
 інтервал\_міри\_колив\_сигма TEXT,  
 інтервал\_міри\_колив\_2\_сигма TEXT,  
 інтервал\_міри\_колив\_3\_сигма TEXT)  
'''**)  
conn.commit()

Оскільки для даних по кожній частині мови в обох вибірках потрібно провести однакові дії, щоб отримати статистичні дані, ці дії було описано за допомогою функцій, які потім було застосовано по черзі до кожної частини мови. Всього таких функцій три:

1. *count\_aver\_freq()* – створює варіаційний ряд абсолютних частот, додає до змінної типу даних list списки з такими величинами: хі,ni, хі\*ni, х̅, хі -х̅, (хі - х̅)2, (хі - х̅,)2 ni, - які потім за допомогою ітерації буде додано до таблиці з відповідною структурою.
2. *statistics()* – на основі створеної таблиць в попередній функції вираховує основні статистичні характеристики:
   1. середнє квадратичне відхилення
   2. довірчий інтервал σ
   3. довірчий інтервал 2σ
   4. довірчий інтервал 3σ
   5. міра коливання середньої частоти
   6. довірчий інтервал σ**х̅**
   7. довірчий інтервал 2σ**х̅**
   8. довірчий інтервал 3σ**х̅**
   9. коефіцієнт варіації
   10. максимальний коефіцієнт варіації
   11. коефіцієнт стабільності
   12. похибка дослідження

Ці значення додаються до змінної типу даних list і пізніше їх буде додано до таблиці з відповідною структурою.

1. *print\_results()* – виводить для кожної частини мови дані з попередніх функцій у консоль у вигляді двох таблиць для кращого сприйняття інформації в процесі створення коду. А також будує полігони частот частин мови з двох вибірок.

Розглянемо функцію *count\_aver\_freq()* більш детально. Виглядає вона таким чином:

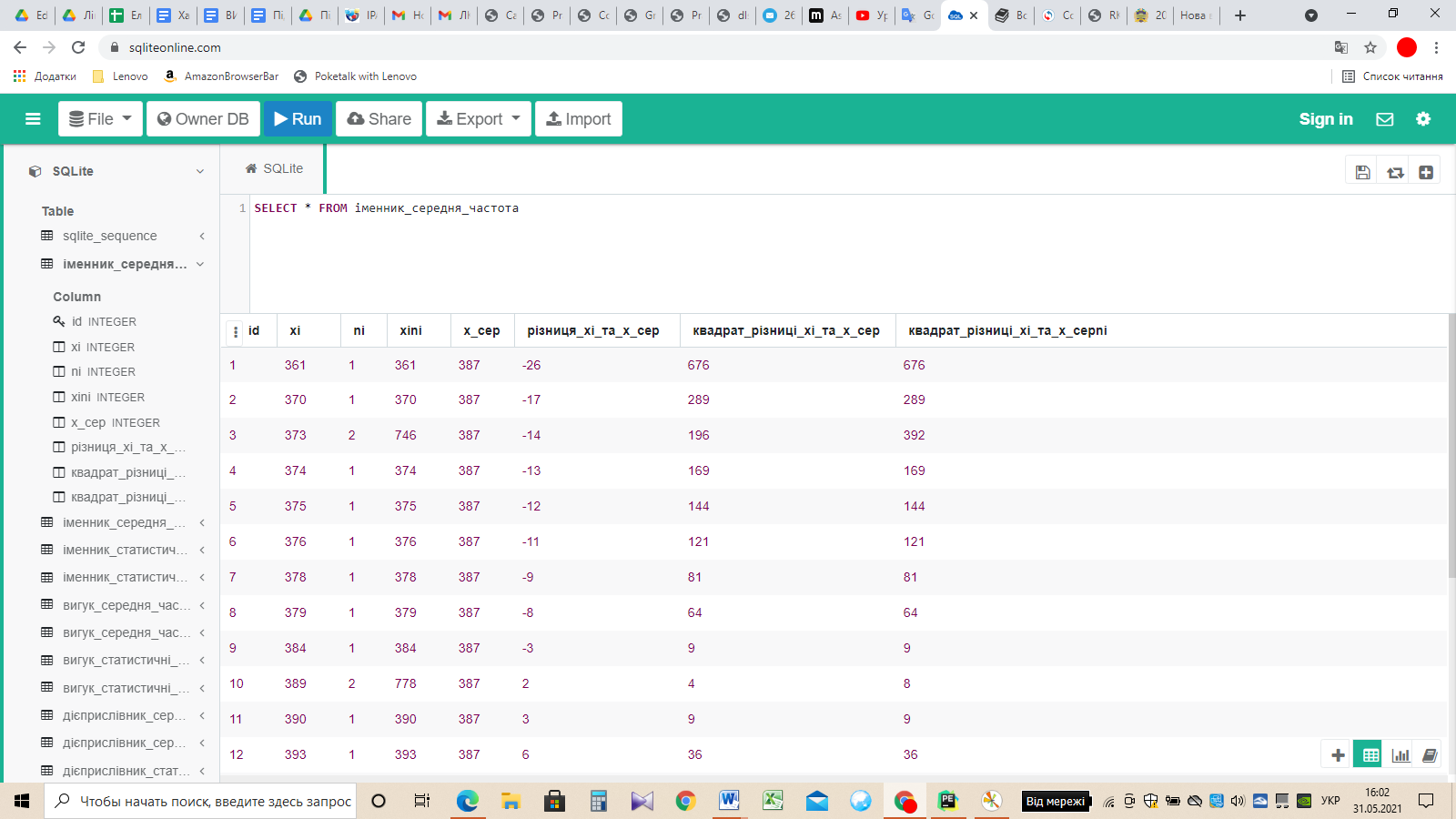
**def** count\_aver\_freq():  
 **global** aver\_freq\_x  
 **global** aver\_freq\_x\_ordered  
 aver\_freq\_x = {}rows = cursor.fetchall()  
 **for** row **in** rows:  
 freq = row[4:24]  
 **for** i **in** freq:  
 **if** i **in** aver\_freq\_x:  
 aver\_freq\_x[i][1] += 1  
 xini0 = aver\_freq\_x[i][0]\*aver\_freq\_x[i][1]  
 aver\_freq\_x[i][2] = xini0  
 **else**:  
 aver\_freq\_x[i] = [i]  
 aver\_freq\_x[i].append(1)  
 xini0 = aver\_freq\_x[i][0]\*aver\_freq\_x[i][1]  
 aver\_freq\_x[i].append(xini0)  
 aver\_freq\_x = list(aver\_freq\_x.values())aver\_freq\_x\_ordered = []**for** i **in** aver\_freq\_x:  
 aver\_freq\_x\_ordered.append(i)**for** i **in** aver\_freq\_x\_ordered:  
 count = 0  
 **while** count **in** range(0, len(aver\_freq\_x\_ordered)-1):  
 **if** aver\_freq\_x\_ordered[count][0]>aver\_freq\_x\_ordered[count+1][0]:  
 aver\_freq\_x\_ordered[count],aver\_freq\_x\_ordered[count+1] = aver\_freq\_x\_ordered[count+1], aver\_freq\_x\_ordered[count]  
 count += 1count=0

**while** count **in** range (0, len(aver\_freq\_x\_ordered)):  
 xini0= []  
 ni0 = []  
 **for** i **in** aver\_freq\_x\_ordered:  
 xini0.append(i[2])  
 ni0.append(i[1])  
 x\_aver0 = sum(xini0)/sum(ni0)  
 aver\_freq\_x\_ordered[count].append(x\_aver0)  
 aver\_freq\_x\_ordered[count].append(round(aver\_freq\_x\_ordered[count][0]-x\_aver0, 2))  
 aver\_freq\_x\_ordered[count].append(round((aver\_freq\_x\_ordered[count][0]-x\_aver0)\*\*2, 2))  
 aver\_freq\_x\_ordered[count].append(round((aver\_freq\_x\_ordered[count][0]-x\_aver0)\*\*2\*aver\_freq\_x\_ordered[count][1], 2))  
 count += 1

Для того, щоб по черзі викликати ці функції для кожної частини мови, спочатку потрібно зчитати з частотного словника рядок, який містить назву частини мови й потім викликати першу функцію. Після цього буде виконано все, що в ній прописано, тобто буде створено список з усіма потрібними даними. Після виклику функції додаємо вміст списку до таблиці \**частина мови\*\_середня\_частота.*

cursor.execute(**"""select \* from част\_частин\_мови  
 where частина\_мови = 'NOUN'"""**)  
count\_aver\_freq() *#for i in aver\_freq\_x\_ordered:  
# cursor.execute("""INSERT INTO іменник\_середня\_частота (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер,  
# квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
# VALUES(?, ?, ?, ?, ?, ?, ?)""", i)  
#conn.commit()*

Таблиця має такий вигляд:

Варіаційний ряд іменника в першій вибірці (таблиця *іменник\_середня\_частота*)

У другій функції – *statistics() –* ми знаходимо всі статистичні дані. Але, оскільки для їх обчислення потрібні значення з таблиці, що була створена після виклику першої функції (хі,ni, х̅, хі -х̅, (хі - х̅)2, (хі - х̅,)2 ni), ми зчитуємо їх з неї:

**def** statistics():  
 **global** xi, ni, xini, ni\_sum, x\_aver, xi\_minus\_x\_aver, xi\_minus\_x\_aver\_squared, xi\_minus\_x\_aver\_squared\_ni, sigma, \  
 sigma\_x\_aver, interval\_sigma, interval\_2\_sigma, interval\_3\_sigma, interval\_sigma\_x\_aver, interval\_sigma\_2\_x\_aver, \  
 interval\_sigma\_3\_x\_aver, coef\_of\_variation, coef\_of\_variation\_max, D, error  
 **global** my\_list  
 rows = cursor.fetchall()  
 xi = []  
 **for** row **in** rows:  
 xi.append(row[1])

У цій частині коду ми зі змінної *rows* додаємо перший елемент кожного елементу до нового списку *хі*. *rows* містить списки, кожен з них представляє рядок з таблиці *частина мови\_середня\_частота.* Перший елемент – це значення хі. Тобто, в результаті ми отримуємо список хі з усіма значеннями абсолютної частоти. Так само ми записуємо в окремий список значення nі , хі\* nі , х̅, , (хі - х̅)2 та (хі - х̅,)2 ni:  
  
 ni = []  
 **for** row **in** rows:  
 ni.append(row[2])  
 ni\_sum = sum(ni)  
  
 xini = []  
 **for** row **in** rows:  
 xini.append(row[3])  
 xini\_sum = sum(xini)x\_aver = rows[0][4]

xi\_minus\_x\_aver = []  
 **for** row **in** rows:  
 xi\_minus\_x\_aver.append(row[5])  
  
 xi\_minus\_x\_aver\_squared = []  
 **for** row **in** rows:  
 xi\_minus\_x\_aver\_squared.append(row[6])  
  
 xi\_minus\_x\_aver\_squared\_ni = []  
 **for** row **in** rows:  
 xi\_minus\_x\_aver\_squared\_ni.append(row[7])

Тепер, коли всі потрібні значення є в функції, можна знайти статистичні характеристики.

1. Сигма (середнє квадратичне відхилення) знаходиться за формулою . В коді її значення було знайдено таким чином:

sigma = sqrt(sum(xi\_minus\_x\_aver\_squared\_ni)/ni\_sum)

1. Міра коливання середньої частоти має формулу . В програмі це значення було знайдено так:sigma\_x\_aver = sigma/sqrt(ni\_sum)
2. Довірчій інтервал для ([ ; ]) було записано в змінну типу даних list. Межі інтервалу було округлено до цілого числа. Знайдено його було таким чином:

interval\_sigma = [round(x\_aver-sigma), round(x\_aver+sigma)]

1. Довірчі інтервали для 2([; ]) і 3 були визначені схожим способом:

interval\_2\_sigma = [round(x\_aver-2\*sigma),round(x\_aver+2\*sigma)]

interval\_3\_sigma = [round(x\_aver-3\*sigma),round(x\_aver+3\*sigma)]

1. І за таким самим принцип було встановлено довірчі інтервали σ**х̅**

([х̅ ; ]), 2σ**х̅** ([х̅ ; ]) і 3 σ**х̅** ([х̅ ; ]):

interval\_sigma\_x\_aver = [round(x\_aver-sigma\_x\_aver),round(x\_aver+sigma\_x\_aver)]  
 interval\_sigma\_2\_x\_aver = [round(x\_aver-2\*sigma\_x\_aver), round(x\_aver+2\*sigma\_x\_aver)]  
 interval\_sigma\_3\_x\_aver = [round(x\_aver-3\*sigma\_x\_aver), round(x\_aver+3\*sigma\_x\_aver)]  
6) Коефіцієнт варіацій дорівнює . Знайдено його було наступним чином:

coef\_of\_variation = round(sigma/x\_aver, 2)

Його значення було округлено до сотень.

1. Максимальний коефіцієнт варіацій має формулу . Знайдено його було таким чином:

coef\_of\_variation\_max = sqrt(ni\_sum-1)

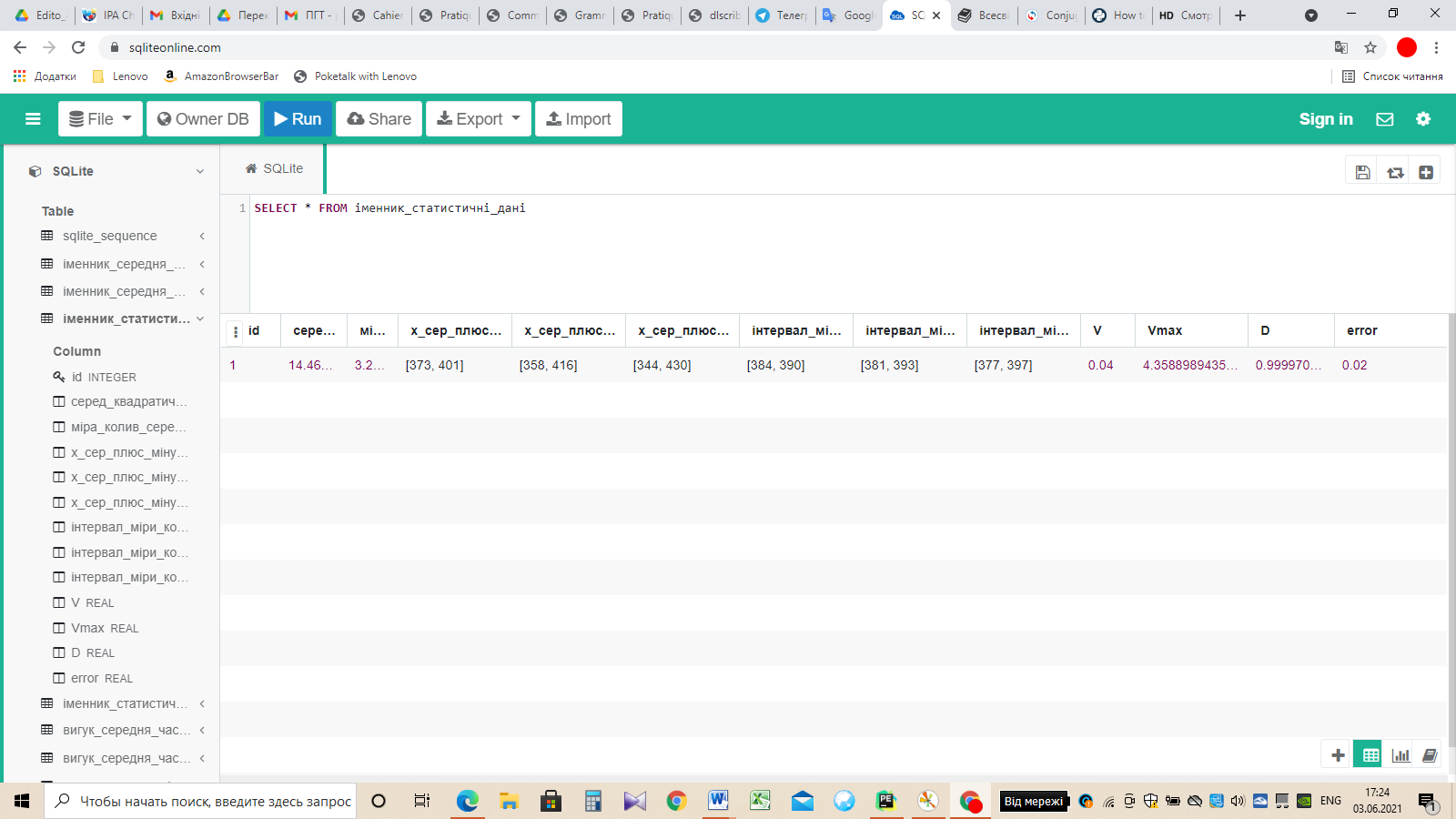
1. Коефіцієнт стабільності) було знайдено таким чином:  
    D = 1-coef\_of\_variation/coef\_of\_variation\_max
2. Похибка дослідження () була вирахувана так:  
    error = round((1.96\*coef\_of\_variation)/sqrt(sum(ni)), 2)

Після цього всі ці значення було додано до змінної my\_list типу даних список. З неї елементи додаватимуться до таблиці в БД.

my\_list = []  
 my\_list.append(sigma)  
 my\_list.append(sigma\_x\_aver)  
 my\_list.append(str(interval\_sigma))  
 my\_list.append(str(interval\_2\_sigma))  
 my\_list.append(str(interval\_3\_sigma))  
 my\_list.append(str(interval\_sigma\_x\_aver))  
 my\_list.append(str(interval\_sigma\_2\_x\_aver))  
 my\_list.append(str(interval\_sigma\_3\_x\_aver))

Її ми викликаємо після формування попередньої таблиці та зчитування з неї даних. Після її виклику всі потрібні характеристики записані в список my\_list, тому наступним кроком ми додаємо його вміст в таблицю *частина мови\_статистичні\_дані.*

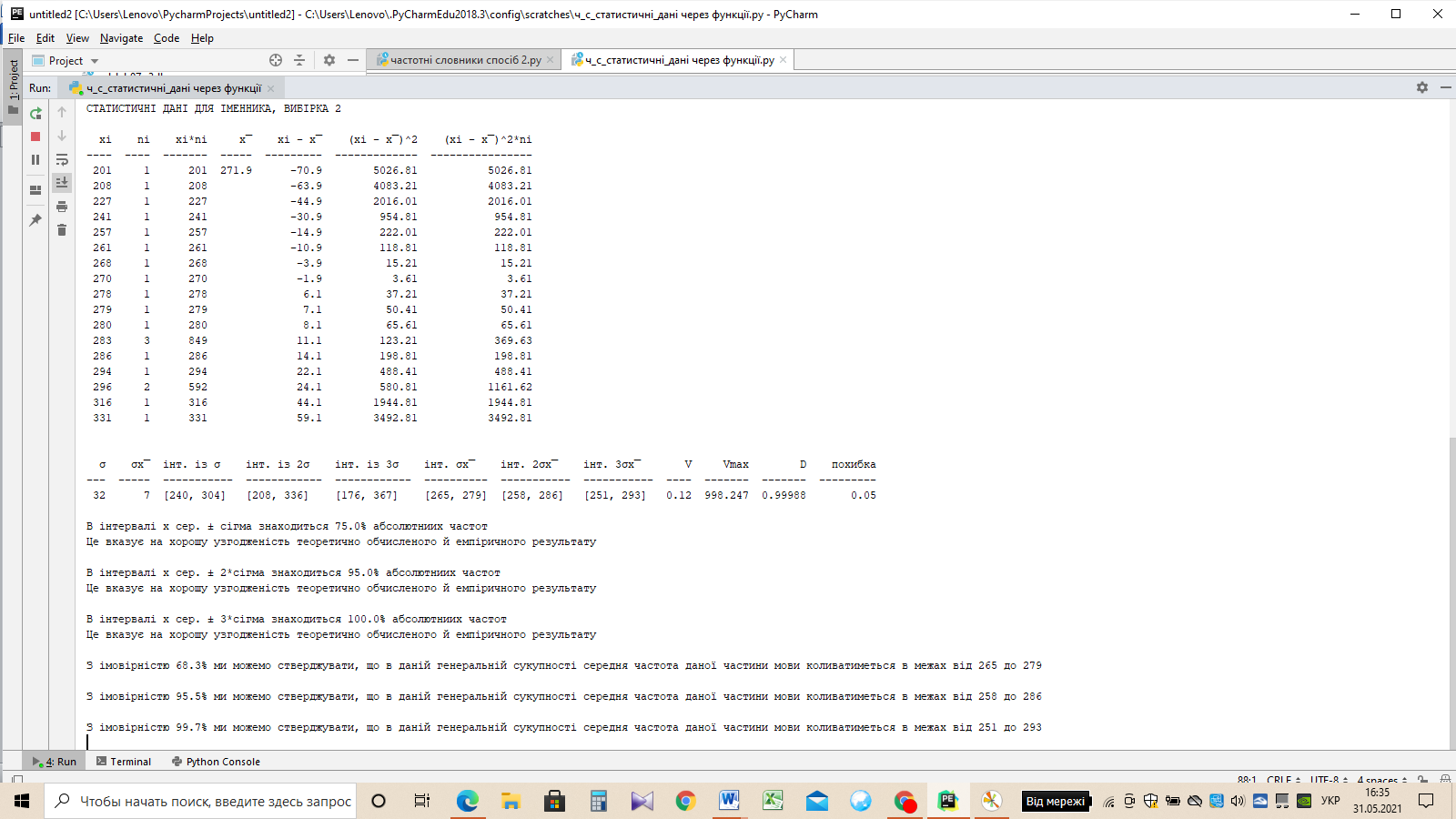
cursor.execute(**"select \* from іменник\_середня\_частота"**)  
statistics()  
cursor.execute(**""" INSERT INTO іменник\_статистичні\_дані (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()

В результаті виклику цієї функції отримуємо таблицю в БД такого вигляду

Таблиця зі статистичними характеристиками іменника в першій вибірці

Третя функція має такий вигляд:

**def** print\_results():  
 table = {**'xi'**: xi,**'ni'**: ni, **'xi\*ni'**: xini, **'х̅'**: [x\_aver], **'xi - х̅'**: xi\_minus\_x\_aver,  
 **'(xi - х̅)^2'**: xi\_minus\_x\_aver\_squared, **'(xi - х̅)^2\*ni'**: xi\_minus\_x\_aver\_squared\_ni}  
 print(tabulate(table, headers=**'keys'**))  
  
 table = [[**'σ'**, **'σх̅'**, **'інт. із σ'**, **'інт. із 2σ'**,  
 **'інт. із 3σ'**, **'інт. σх̅'**, **'інт. 2σх̅'**,  
 **'інт. 3σх̅'**, **'V'**, **'Vmax'**, **'D'**, **'похибка'**], [round(sigma), round(sigma\_x\_aver),  
 interval\_sigma, interval\_2\_sigma, interval\_3\_sigma, interval\_sigma\_x\_aver, interval\_sigma\_2\_x\_aver,  
 interval\_sigma\_3\_x\_aver, coef\_of\_variation, coef\_of\_variation\_max, D, error]]  
 print(**'\n'**)  
 print(tabulate(table, headers=**'firstrow'**))  
  
  
 x\_aver\_minus\_sigma = round(interval\_sigma[0])  
 x\_aver\_plus\_sigma = round(interval\_sigma[1])  
 *#print(x\_aver\_minus\_sigma)  
 #print(x\_aver\_plus\_sigma)* **global** ni\_in\_interval\_sigma\_sum  
 ni\_in\_interval\_sigma = []  
 **for** i **in** aver\_freq\_x\_ordered:  
 **if** i[0] **in** range(x\_aver\_minus\_sigma, x\_aver\_plus\_sigma+1):  
 ni\_in\_interval\_sigma.append(i[1])  
 ni\_in\_interval\_sigma\_sum = sum(ni\_in\_interval\_sigma)  
 percentage = round(ni\_in\_interval\_sigma\_sum\*100/ni\_sum, 1)  
  
 print(**'\n'** + **'В інтервалі x сер. ± сігма знаходиться '** + str(percentage) + **'% абсолютниих частот'**)  
  
 **if** round((68.3 - percentage)\*100/68.3, 1) < 5:  
 print(**'Це вказує на хорошу узгодженіcть теоретично обчисленого й емпіричного результату'**)  
 **else**:  
 print(**'Це вказує на погану узгодженіcть теоретично обчисленого й емпіричного результату'**)  
  
 x\_aver\_minus\_2\_sigma = round(interval\_2\_sigma[0])  
 x\_aver\_plus\_2\_sigma = round(interval\_2\_sigma[1])**global** ni\_in\_interval\_2\_sigma\_sum  
 ni\_in\_interval\_2\_sigma = []  
 **for** i **in** aver\_freq\_x\_ordered:  
 **if** i[0] **in** range(x\_aver\_minus\_2\_sigma, x\_aver\_plus\_2\_sigma+1):  
 ni\_in\_interval\_2\_sigma.append(i[1])  
 ni\_in\_interval\_2\_sigma\_sum = sum(ni\_in\_interval\_2\_sigma)  
 percentage = round(ni\_in\_interval\_2\_sigma\_sum\*100/ni\_sum, 1)  
 print(**'\n'** + **'В інтервалі x сер. ± 2\*сігма знаходиться '** + str(percentage) + **'% абсолютниих частот'**)  
  
 **if** round((95.5 - percentage)\*100/95.5, 1) < 5:  
 print(**'Це вказує на хорошу узгодженіcть теоретично обчисленого й емпіричного результату'**)  
 **else**:  
 print(**'Це вказує на погану узгодженіcть теоретично обчисленого й емпіричного результату'**)  
  
 x\_aver\_minus\_3\_sigma = round(interval\_3\_sigma[0])  
 x\_aver\_plus\_3\_sigma = round(interval\_3\_sigma[1])**global** ni\_in\_interval\_3\_sigma\_sum  
 ni\_in\_interval\_3\_sigma = []  
 **for** i **in** aver\_freq\_x\_ordered:  
 **if** i[0] **in** range(x\_aver\_minus\_3\_sigma, x\_aver\_plus\_3\_sigma+1):  
 ni\_in\_interval\_3\_sigma.append(i[1])  
 ni\_in\_interval\_3\_sigma\_sum = sum(ni\_in\_interval\_3\_sigma)  
 percentage = round(ni\_in\_interval\_3\_sigma\_sum\*100/ni\_sum, 1)  
 print(**'\n'** + **'В інтервалі x сер. ± 3\*сігма знаходиться '** + str(percentage) + **'% абсолютних частот'**)  
  
 **if** round((99.7 - percentage)\*100/99.7, 1) < 5:  
 print(**'Це вказує на хорошу узгодженість теоретично обчисленого й емпіричного результату'**)  
 **else**:  
 print(**'Це вказує на погану узгодженість теоретично обчисленого й емпіричного результату'**)  
  
 x\_aver\_minus\_sigma\_x\_aver = round(interval\_sigma\_x\_aver[0])  
 x\_aver\_plus\_sigma\_x\_aver = round(interval\_sigma\_x\_aver[1])  
 print(**'\n'** + **'''З імовірністю 68.3% ми можемо стверджувати, що в даній генеральній сукупності середня частота даної частини мови коливатиметься в межах '''** + **'від '**+ str(x\_aver\_minus\_sigma\_x\_aver) + **' до '** + str(x\_aver\_plus\_sigma\_x\_aver))  
  
 x\_aver\_minus\_2\_sigma\_x\_aver = round(interval\_sigma\_2\_x\_aver[0])  
 x\_aver\_plus\_2\_sigma\_x\_aver = round(interval\_sigma\_2\_x\_aver[1])  
 print(**'\n'** + **'''З імовірністю 95.5% ми можемо стверджувати, що в даній генеральній сукупності середня частота даної частини мови коливатиметься в межах '''** + **'від '**+ str(x\_aver\_minus\_2\_sigma\_x\_aver) + **' до '** + str(x\_aver\_plus\_2\_sigma\_x\_aver))  
  
 x\_aver\_minus\_3\_sigma\_x\_aver = round(interval\_sigma\_3\_x\_aver[0])  
 x\_aver\_plus\_3\_sigma\_x\_aver = round(interval\_sigma\_3\_x\_aver[1])  
 print(**'\n'** + **'''З імовірністю 99.7% ми можемо стверджувати, що в даній генеральній сукупності середня частота даної частини мови коливатиметься в межах '''** + **'від '**+ str(x\_aver\_minus\_3\_sigma\_x\_aver) + **' до '** + str(x\_aver\_plus\_3\_sigma\_x\_aver))

Після виклику цієї функції всі статистичні дані відображаються в консолі в такому вигляді:

**Отримані статистичні характеристики**

## Іменник

**1 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 14 | [358, 416] | 3 | [381, 393] | 0.04 | 4.3589 | 0.99 | 0.02 |

**2 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 32 | [208, 336] | 7 | [258, 286] | 0.12 | 4.3589 | 0.97 | 0.05 |

Отже, в першій вибірці абсолютні частоти іменника можуть відхилятися від середньої частоти на 14, а в другій – на 32.

В довірчому інтервалі **±** σ в першій вибірці знаходиться 12 (60%) абсолютних частот іменника. Це вказує на погану узгодженіcть теоретично обчисленого й емпіричного результату.

В довірчому інтервалі з **±** σ в другій вибірці знаходиться 15 (75%) абсолютних частот іменника. Це вказує на гарну узгодженіcть теоретично обчисленого й емпіричного результату.

В довірчому інтервалі з **±** 2σ в першій вибірці знаходиться 20 (100%) абсолютних частот іменника. Це вказує на хорошу узгодженіcть теоретично обчисленого й емпіричного результату. В другій вибірці на цей інтервал припадає знаходиться 19 (95%) абсолютних частот іменника, що також вказує на хорошу узгодженіcть теоретично обчисленого й емпіричного результату.

Графічні смуги коливання частоти іменника в обох вибірках при довірчій імовірності 99% не перетинаються, отже можна стверджувати, що вибірки належать до різних генеральних сукупностей. Отже, нульова гіпотеза відкидається.

Міра коливання середньої частоти іменника в першій вибірці в генеральній сукупності становить 3. При довірчому інтервалі **±** 2σх̅ з імовірністю 95.5% можна стверджувати, що в генеральній сукупності середня частота іменника коливатиметься в межах від 358 до 416. Міра коливання середньої частоти іменника в генеральній сукупності другій вибірці становить в 7. При довірчому інтервалі **±** 2σх̅ з імовірністю 95.5% можна стверджувати, що в генеральній сукупності середня частота іменника коливатиметься в межах від 208 до 336.

Коефіцієнт варіацій іменника в першій вибірці становить 4%, а в другій – 12%, що вказує на те, що стабільність уживання іменника в першій вибірці більша, ніж в другій.

Коефіцієнт стабільності в першій вибірці для іменника високий (99%), а в другій – 97%. Отже рівномірність розподілу іменника в обох вибірках висока, але в першій ця частина мови розподілена більш рівномірно.

Відносна похибка дослідження в першій вибірці становить 2%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 2%. За першою прикидкою δ дорівнювала також 2%.

Відносна похибка дослідження в другій становить 5%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 5%. За першою прикидкою δ дорівнювала 2.7%.

## Дієслово

**1 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 20 | [83, 162] | 4 | [113, 131] | 0.16 | 4.3589 | 0.96 | 0.07 |

**2 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 19 | [127, 205] | 4 | [157, 175] | 0.12 | 4.3589 | 0.97 | 0.05 |

1. Отже, в першій вибірці абсолютні частоти дієслова можуть відхилятися від середньої частоти на 20, а в другій – на 19.

В обох вибірках в довірчому інтервалі **±** σ знаходиться 15 (75%) абсолютних частот дієслова. Це вказує на хорошу узгодженіcть теоретично обчисленого й емпіричного результату.

В обох вибірках в інтервалі **±** 2σ знаходиться 19 (95%) абсолютних частот дієслова, що вказує на хорошу узгодженіcть теоретично обчисленого й емпіричного результату.

Графічні смуги коливання частоти дієслова в обох вибірках при довірчій імовірності 99% не перетинаються, отже можна стверджувати, що вибірки належать до різних генеральних сукупностей. Отже, нульова гіпотеза відкидається.

Міра коливання середньої частоти в генеральній сукупності в першій і другій вибірках такж збігається і становить 4. При довірчому інтервалі **±** 2σх̅ з імовірністю 95.5% можна стверджувати, що в генеральній сукупності в першій вибірці середня частота дієслова коливатиметься в межах від 113 до 131, а в другій - від 157 до 175.

Коефіцієнт варіацій в першій вибірці становить 16%, а в другій – 12 ,отже, стабільність уживання дієслова в першій вибірці менша, ніж в другій.

Коефіцієнт стабільності в першій вибірці для дієслова високий (96%), отже рівномірність розподілу дієслова в ній висока. В другій вибірці цей показник вищий (97%), отже рівномірність розподілу дієслова в ній більша, ніж в першій.

Відносна похибка дослідження в першій вибірці становить 7%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 7%. В другій вибірці відносна похибка дослідження становить 5%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 5%.

## Прикметник

**1 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 25 | [99, 199] | 6 | [138, 160] | 0.17 | 4.3589 | 0.96 | 0.07 |

**2 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 19 | [62, 136] | 4 | [91, 107] | 0.19 | 4.3589 | 0.96 | 0.08 |

В першій вибірці абсолютні частоти прикметника можуть відхилятися від середньої частоти на 25, в той час, як в другій на 19.

В довірчому інтервалі **±** σ в першій вибірці знаходиться 13 (65%) абсолютних частот. Це вказує на не дуже гарну узгодженіcть теоретично обчисленого й емпіричного результату. А от в другій вибірці в такому довірчому інтервалі знаходиться 16 (80%) абсолютних частот, що вказує на гарну узгодженіcть теоретично обчисленого й емпіричного результату.

В довірчому інтервалі **±** 2σ в першій вибірці знаходиться 18 (90%) абсолютних частот прикметника, а в другій - 20 (100%). Це вказує на не дуже хорошу узгодженіcть теоретично обчисленого й емпіричного результату.

Графічні смуги коливання частоти прикметника в обох вибірках при довірчій імовірності 99% не перетинаються, отже можна стверджувати, що вибірки належать до різних генеральних сукупностей. Отже, нульова гіпотеза відкидається.

Міра коливання середньої частоти в генеральній сукупності вибіркок становить 6 і 4 відповідно. При довірчому інтервалі **±** 2σх̅ на основі дослідження першої вибіркиз імовірністю 95.5% можна стверджувати, що в генеральній сукупності середня частота цієї частини мови коливатиметься в межах від 138 до 160. У генеральній сукупності другої вибірки з такою ж імовірністю можна стверджувати, що середня частота прикметника коливатиметься в межах від 91 до 107.

Коефіцієнт варіацій в першій вибірцістановить 17%, а в другій – 19%, що вказує на те, що стабільність уживання прикметника в першій вибірці більша, ніж в другій.

Коефіцієнт стабільності в вибірках для прикметника однаковий і дорівнює 96%, отже рівномірність розподілу прикметника в них висока.

Відносна похибка дослідження в вибірках становить 7% і 8% відповідно, отже, з ймовірністю 95% можна стверджувати, що значення для прикметника в першій вибірці обчислено з відносною погрішністю 7%, а в другій з погрішністю у 8%.

## Прислівник

**1 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 6 | [25, 49] | 1 | [34, 39] | 0.16 | 4.3589 | 0.96 | 0.07 |

**2 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 9 | [48, 86] | 2 | [63, 71] | 0.14 | 4.3589 | 0.97 | 0.06 |

Отже, в першій вибірці абсолютні частоти можуть відхилятися від середньої частоти на 6, а в другій – на 9.

В довірчому інтервалі **±** σ в першій вибірці знаходиться 15 (75%) абсолютних частот. Це вказує на гарну узгодженіcть теоретично обчисленого й емпіричного результату. У другій же вибірцці на цей довірчій інтервал припадає 12 (60%) абсолютних частот прислівника. Це вказує на погану узгодженіcть теоретично обчисленого й емпіричного результату.

В першій вибірці в довірчому інтервалі **±** 2σ знаходиться 19 (95%) абсолютних частот прислівника, а в другій - 20 (100%). Це вказує на хорошу узгодженіcть теоретично обчисленого й емпіричного результату.

Графічні смуги коливання частоти прислівника в обох вибірках при довірчій імовірності 99% не перетинаються, отже можна стверджувати, що вибірки належать до різних генеральних сукупностей. Отже, нульова гіпотеза відкидається.

Міра коливання середньої частоти в генеральній сукупності першої вибірки становить 1, а другої – 2. При довірчому інтервалі **±** 2σх̅ з імовірністю 95.5% можна стверджувати, що в генеральній сукупності першої вибірки середня частота цієї частини мови коливатиметься в межах від 34 до 39, і в межах від 63 до 71в генеральній сукупності другої вибірки.

Коефіцієнт варіацій в підвибірках становить 16%, і 14% що вказує на те, що стабільність уживання прислівника в першій вибірці менша, ніж в другій.

Коефіцієнт стабільності в другій вибірці становить 97% і вищий за цей показник в першій вибірці (96%). Рівномірність розподілу прислівника в обох вибірках висока.

Відносна похибка дослідження в першій вибірці становить 7%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 7%. В другій вибірці значення обчислено з відносною погрішністю 6%.

## Числівник

**1 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | [-1, 4] | 0 | [1, 2] | 0.99 | 4.3589 | 0.64 | 0.43 |

**2 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 3 | [-1, 9] | 1 | [3, 5] | 0.68 | 4.3589 | 0.90 | 0.3 |

В першій вибірці абсолютні частоти числівника можуть відхилятися від середньої частоти на 1, а в другій – на 3.

В довірчому інтервалі **±** σ в першій вибірці знаходиться 19 (95%), в другій - 17 (85%) абсолютних частот числівника. Це вказує на гарну узгодженіcть теоретично обчисленого й емпіричного результату.

В довірчому інтервалі **±** 2σ в обох вибірках знаходиться 19 (95%) абсолютних частот числівника. Це теж вказує на хорошу узгодженіcть теоретично обчисленого й емпіричного результату.

Міра коливання середньої частоти в генеральній сукупності першої вибірки становить 0. При довірчому інтервалі **±** 2σх̅ з імовірністю 95.5% можна стверджувати, що в генеральній сукупності середня частота цієї частини мови коливатиметься в межах від 1 до 2. Міра коливання середньої частоти числівника в генеральній сукупності другої вибірки становить 1. При довірчому інтервалі **±** 2σх̅ з імовірністю 95.5% можна стверджувати, що в генеральній сукупності середня частота частини мови коливатиметься в межах від 3 до 5.

Коефіцієнт варіацій становить 99%, що вказує на те, що середнє квадратичне відхилення становить 99% від середньої частоти та, що стабільність уживання прислівника в цій вибірці менша, ніж в другій, бо в другій вибірці він дорівнює 68%.

Коефіцієнт стабільності в першій вибірці для числівника невисокий (64%), отже рівномірність розподілу числівника в ній низька. В другій вибірці цей показник набагато вищий (90%), отже, рівномірність розподілу числівника в ній висока.

Відносна похибка дослідження в першій вибірці становить 43%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 43%. А значить, обсяг вибірки не достатній, щоб проводити статистичне дослідження цієї частини мови. В другій вибірці відносна похибка дослідження становить 3%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 3%.

## Займенник

**1 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 13 | [52, 104] | 3 | [72, 84] | 0.16 | 4.3589 | 0.96 | 0.07 |

**2 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 18 | [99, 173] | 4 | [128, 144] | 0.13 | 4.3589 | 0.97 | 0.06 |

Абсолютні частоти займенника в першій вибірці можуть відхилятися від середньої частоти на 13, а в другій – на 18.

В першій вибірці в довірчому інтервалі **±** σ знаходиться 15 (75%) абсолютних частот, а в другій – 14 (70%). Це вказує на гарну узгодженіcть теоретично обчисленого й емпіричного результату.

В довірчому інтервалі **±** 2σ в першій вибірці знаходиться 18 (90%) абсолютних частот займенника, що вказує на не дуже хорошу узгодженіcть теоретично обчисленого й емпіричного результату. В другій вибірці на цей довірчій інтервал припадає 19 (95%) абсолютних частот, отже узгодженіcть теоретично обчисленого й емпіричного результату гарна.

Графічні смуги коливання частоти займенника в обох вибірках при довірчій імовірності 99% не перетинаються, отже можна стверджувати, що вибірки належать до різних генеральних сукупностей. Отже, нульова гіпотеза відкидається.

Міра коливання середньої частоти займенника в генеральній сукупності першої вибірки становить 3. При довірчому інтервалі **±** 2σх̅ з імовірністю 95.5% можна стверджувати, що в генеральній сукупності середня частота цієї частини мови коливатиметься в межах від 72 до 84. Міра коливання середньої частоти в генеральній сукупності другої вибірки становить 4. При такому ж довірчому інтервалі з імовірністю 95.5% можна стверджувати, що в генеральній сукупності середня частота займенника коливатиметься в межах від 128 до 144.

Коефіцієнт варіацій в першій вибірці становить 16%, що вказує на те, що що стабільність уживання займенника в цій вибірці менша, ніж в другій (13%).

Коефіцієнт стабільності в першій вибірці для займенника дорівнює 96%, а в другій – 97%, отже, рівномірність розподілу займенника в них висока.

Відносна похибка дослідження в першій вибірці становить 7%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 7%. В другій вибірці цей показник становить 6%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 6%.

## Прийменник

**1 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 9 | [46, 83] | 2 | [60, 69] | 0.14 | 4.3589 | 0.97 | 0.06 |

**2 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 9 | [33, 69] | 2 | [47, 55] | 0.18 | 4.3589 | 0.96 | 0.08 |

Отже, в обох вибірках абсолютні частоти прийменника можуть відхилятися від середньої частоти на 9.

В довірчому інтервалі **±** σ в першій вибірці знаходиться 13 (65%) абсолютних частот. Це вказує на доволі гарну узгодженіcть теоретично обчисленого й емпіричного результату. В другій вибірці на цей довірчій інтервал припадає 16 (80%) абсолютних частот прийменника, що вказує на гарну узгодженіcть теоретично обчисленого й емпіричного результату.

В довірчому інтервалі **±** 2σ в першій вибірці знаходиться 19 (95%) абсолютних частот прийменника. Це вказує на хорошу узгодженіcть теоретично обчисленого й емпіричного результату. В другій вибірці в цьому інтервалі 2σ знаходиться 18 (90%) абсолютних частот. Це вказує на не дуже хорошу узгодженіcть теоретично обчисленого й емпіричного результату.

Графічні смуги коливання частоти прийменника в обох вибірках при довірчій імовірності 99% не перетинаються, отже можна стверджувати, що вибірки належать до різних генеральних сукупностей. Отже, нульова гіпотеза відкидається.

Міра коливання середньої частоти в генеральній сукупності обох вибірок становить 2. При довірчому інтервалі **±** 2σх̅ з імовірністю 95.5% можна стверджувати, що в генеральній сукупності першої вибірки середня частота цієї частини мови коливатиметься в межах від 60 до 69, а другої – в межах від 47 до 55 .

Коефіцієнт варіацій становить 14% і 18% у вибірках, отже, стабільність уживання прийменника в першій вибірці більша, ніж в другій.

Коефіцієнт стабільності в першій вибірці для прийменника (97%) вищий, ніж в другій (96%). Але в обох вибірках рівномірність розподілу прийменника висока.

В першій вибірці відносна похибка дослідження становить 6%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 6%. Відносна похибка дослідження в другій вибірці становить 8%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 8%.

## Сполучник

**1 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 10 | [61, 100] | 2 | [76, 85] | 0.12 | 4.3589 | 0.97 | 0.05 |

**2 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 10 | [60, 102] | 2 | [76, 86] | 0.13 | 4.3589 | 0.97 | 0.06 |

Абсолютні частоти сполучника в обох вибірках можуть відхилятися від середньої частоти на 10.

В довірчому інтервалі **±** σ в першій вибірці знаходиться 12 (60%) абсолютних частот, а в другій – 10 (50%). Це вказує на погану узгодженіcть теоретично обчисленого й емпіричного результату.

В довірчому інтервалі **±** 2σ в обох вибірках знаходиться 20 (100%) абсолютних частот сполучника. Це вказує на хорошу узгодженіcть теоретично обчисленого й емпіричного результату.

Графічні смуги коливання частоти сполучника в обох вибірках при довірчих імовірностях 68%, 95% і 99% перетинаються, отже нульова гіпотеза відносно цієї частини мови приймається.

Міра коливання середньої частоти в генеральній сукупності обох вибірок становить 2. При довірчому інтервалі **±** 2σх̅ з імовірністю 95.5% можна стверджувати, що в генеральній сукупності першої середня частота сполучника коливатиметься в межах від 76 до 85, а другої - в межах від 76 до 86.

Коефіцієнт варіацій в першій вибірці становить 12%, а в другій – 13%, що вказує на те, що стабільність уживання сполучника в першій вибірці більша, ніж в другій.

Коефіцієнт стабільності в обох вибірках для сполучника високий (97%), отже рівномірність розподілу сполучника в них висока.

Відносна похибка дослідження в першій вибірці становить 5%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 5%. В другій вибірці відносна похибка дослідження становить 6%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 6%.

## Частка

**1 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 10 | [23, 63] | 2 | [39, 48] | 0.23 | 4.3589 | 0.95 | 0.1 |

**2 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 15 | [53, 111] | 3 | [75, 89] | 0.18 | 4.3589 | 0.96 | 0.08 |

Отже, в першій вибірці абсолютні частоти частки можуть відхилятися від середньої частоти на 10, а в другій – на 15.

В довірчому інтервалі **±** σ в першій вибірці знаходиться 15 (75%) абсолютних частот. В другій вибірці до цього довірчого інтервалу потрапляє 14 (70%) абсолютних частот . Це вказує на хорошу узгодженіcть теоретично обчисленого й емпіричного результату.

В довірчому інтервалі **±** 2σ в обох вибірках знаходиться 19 (95%) абсолютних частот частки. Це вказує на хорошу узгодженіcть теоретично обчисленого й емпіричного результату.

Графічні смуги коливання частоти частки в обох вибірках при довірчій імовірності 99% не перетинаються, отже можна стверджувати, що вибірки належать до різних генеральних сукупностей. Отже, нульова гіпотеза відкидається.

Міра коливання середньої частоти в генеральній сукупності становить 2 в першій вибірці і 3 в другій. При довірчому інтервалі **±** 2σх̅ з імовірністю 95.5% можна стверджувати, що в генеральній сукупності першої середня частота цієї частини мови коливатиметься в межах від 39 до 48. В генеральній сукупності другої вибірки при такому ж довірчому інтервалі середня частота частки коливатиметься в межах від 75 до 89.

Коефіцієнт варіацій в першій вибірці становить 23%, і 18% - в другій, що вказує на те, що стабільність уживання частки в першій вибірці менша, ніж в другій.

Коефіцієнт стабільності в першій вибірці для частки становить 95%, а в другій – 96%, отже рівномірність розподілу частки в них висока.

Відносна похибка дослідження в першій вибірці становить 10%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 10%. Відносна похибка дослідження в другій вибірці становить 8%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 8%.

## Дієприслівник

**1 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | [0, 6] | 0 | [2, 3] | 0.56 | 4.3589 | 0.87 | 0.25 |

**2 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 3 | [4, 15] | 1 | [8, 11] | 0.31 | 4.3589 | 0.93 | 0.14 |

Отже, в першій вибірці абсолютні частоти дієприслівника можуть відхилятися від середньої частоти на 2, а в другій – на 3.

В обоїх вибірках в довірчому інтервалі **±** σ знаходиться 15 (75%) абсолютних частот. Це вказує на хорошу узгодженіcть теоретично обчисленого й емпіричного результату.

В довірчому інтервалі **±** 2σ знаходиться 20 (100%) абсолютних частот дієприслівника. Це вказує на хорошу узгодженіcть теоретично обчисленого й емпіричного результату. А от в другій вибірці цей довірчій інтервал містить тільки 18 (90%) абсолютних частот, що вказує на не дуже хорошу узгодженіcть теоретично обчисленого й емпіричного результату.

Графічні смуги коливання частоти дієприслівника в обох вибірках при довірчій імовірності 99% не перетинаються, отже можна стверджувати, що вибірки належать до різних генеральних сукупностей. Отже, нульова гіпотеза відкидається.

Міра коливання середньої частоти в генеральній сукупності першої вибірки становить 0, а другої - 1. При довірчому інтервалі **±** 2σх̅ з імовірністю 95.5% можна стверджувати, що в генеральній сукупності першої вибіркисередня частота дієприслівника коливатиметься в межах від 2 до 3, а в генеральній сукупності другої - в межах від 8 до 11.

Коефіцієнт варіацій в першій вибірці становить 56, отже, стабільність вживання частки в ній низька. В другій вибірці вона значно вища, адже коефіцієнт варіацій в ній становить 31%.

Коефіцієнт стабільності в першій вибірці для дієприслівника достатньо високий (87%), отже рівномірність розподілу дієприслівника в ній досить висока. Коефіцієнт стабільності в другій вибірці вищий (93%).

В першій вибірці відносна похибка дослідження становить 25%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 25%. В другій вибірці цей показник становить 14%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 14%.

## Компаратив

**1 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | [0, 4] | 0 | [2, 3] | 0.59 | 4.3589 | 0.86 | 0.26 |

**2 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | [1, 10] | 1 | [4, 6] | 0.45 | 4.3589 | 0.90 | 0.2 |

Отже, в першій вибірці абсолютні частоти компаратива можуть відхилятися від середньої частоти на 1, а в другій – на 2.

В першій вибірці в довірчому інтервалі **±** σ знаходиться 16 (80%) абсолютних частот, а в другій – 17(85%). Це вказує на хорошу узгодженіcть теоретично обчисленого й емпіричного результату.

В довірчому інтервалі **±** 2σ в обох вибірках знаходиться 20 (100%) абсолютних частот компаратива. Це вказує на хорошу узгодженіcть теоретично обчисленого й емпіричного результату.

Графічні смуги коливання частоти компаратива в обох вибірках при довірчій імовірності 99% не перетинаються, отже можна стверджувати, що вибірки належать до різних генеральних сукупностей. Отже, нульова гіпотеза відкидається.

Міра коливання середньої частоти в генеральній сукупності першої вибірки становить 0, а другої - 1. При довірчому інтервалі **±** 2σх̅ з імовірністю 95.5% можна стверджувати, що в генеральній сукупності першої вибіркисередня частота компаратива коливатиметься в межах від 2 до 3, а в генеральній сукупності другої - в межах від 4 до 6.

Коефіцієнт варіацій в першій вибірці становить 59, отже, стабільність вживання компаратива в ній низька. В другій вибірці вона значно вища, адже коефіцієнт варіацій в ній становить 45%.

Коефіцієнт стабільності в першій вибірці для компаратива достатньо високий (86%), отже рівномірність розподілу компаратива в ній досить висока. Коефіцієнт стабільності в другій вибірці вищий (90%).

В першій вибірці відносна похибка дослідження становить 26%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 26%. В другій вибірці цей показник становить 20%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 20%.

## Вигук

**1 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 5 | [12, 31] | 1 | [19, 24] | 0.22 | 4.3589 | 0.95 | 0.1 |

**2 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 5 | [10, 30] | 1 | [18, 22] | 0.24 | 4.3589 | 0.95 | 0.11 |

Абсолютні частоти вигука в обох вибірках можуть відхилятися від середньої частоти на 5.

В першій вибірці в довірчому інтервалі **±** σ знаходиться 17 (85%) абсолютних частот, а в другій – 15 (75%). Це вказує на гарну узгодженіcть теоретично обчисленого й емпіричного результату.

В довірчому інтервалі **±** 2σ в першій вибірці знаходиться 19 (95%) . В другій вибірці на цей довірчій інтервал припадає 20 (100%) абсолютних частот, отже узгодженіcть теоретично обчисленого й емпіричного результату гарна.

Графічні смуги коливання частоти вигука в обох вибірках при довірчій імовірності 99% перетинаються, отже нульова гіпотеза відносно цієї частини мови приймається.

Міра коливання середньої частоти вигука в генеральній сукупності обох вибірок становить 1. При довірчому інтервалі **±** 2σх̅ з імовірністю 95.5% можна стверджувати, що в генеральній сукупності першої вибірки середня частота цієї частини мови коливатиметься в межах від 19 до 24. А в генеральній сукупності другої – в межах від 18 до 22.

Коефіцієнт варіацій в першій вибірці становить 22%, що вказує на те, що що стабільність уживання вигука в цій вибірці більша, ніж в другій (24%).

Коефіцієнт стабільності для вигука в обох вибірках дорівнює 95%, , отже, рівномірність розподілу вигука в них висока.

Відносна похибка дослідження в першій вибірці становить 10%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 10%. В другій вибірці цей показник становить 11%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 11%.

## Предикатив

**1 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | [-1, 5] | 0 | [1, 3] | 0.84 | 4.3589 | 0.81 | 0.37 |

**2 вибірка**

| σ | інт. із 2σ | σх̅ | інт. 2σх̅ | υ | υmax | D | ε |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | [-1, 7] | 0 | [2, 4] | 0.76 | 4.3589 | 0.83 | 0.33 |

Абсолютні частоти предикатива в обох вибірках можуть відхилятися від середньої частоти на 2.

В першій вибірці в довірчому інтервалі **±** σ знаходиться 19 (95%) абсолютних частот, а в другій – 18 (90%). Це вказує на гарну узгодженіcть теоретично обчисленого й емпіричного результату.

В довірчому інтервалі **±** 2σ в обох вибірках знаходиться 19 (95%), отже узгодженіcть теоретично обчисленого й емпіричного результату гарна.

Графічні смуги коливання частоти предикатива в обох вибірках при довірчій імовірності 99% перетинаються.

Міра коливання середньої частоти вигука в генеральній сукупності обох вибірок становить 0. При довірчому інтервалі **±** 2σх̅ з імовірністю 95.5% можна стверджувати, що в генеральній сукупності першої вибірки середня частота цієї частини мови коливатиметься в межах від 1 до 3. А в генеральній сукупності другої – в межах від 2 до 4.

Коефіцієнт варіацій в першій вибірці становить 84%, що вказує на те, що що стабільність уживання предикатива в цій вибірці менша, ніж в другій (76%), і, що стабільність вживання предикатива в вибірках дуже низька.

Коефіцієнт стабільності для предикатива в першій вибірці дорівнює 81%, а в другій – 83%.

Відносна похибка дослідження в першій вибірці становить 37%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 37%. В другій вибірці цей показник становить 33%, отже з ймовірністю 95% можна стверджувати, що значення обчислено з відносною погрішністю 33%. Отже, відносна похибка дослідження для предикатива занадто висока, тому робити висновки на його дослідженні не можна.

# РОЗДІЛ 3. Зіставлення статистичних характеристик

| **Частина мови** | **Критерій Стьюдента** | **χ2** |
| --- | --- | --- |
| **іменник** | 14.35454 | 53.82163 |
| **дієслово** | 17.42863 | 33.71546 |
| **прикметник** | 2.36499 | 32.58266 |
| **сполучник** | 3.91792 | 23.02739 |
| **займенник** | 0.72858 | 63.89939 |
| **прийменник** | 15.25259 | 21.55376 |
| **частка** | 2.49864 | 59.63968 |
| **прислівник** | 12.47488 | 25.11084 |
| **вигук** | 18.99153 | 19.20751 |
| **компаратив** | 15.82043 | 13.78763 |
| **дієприслівник** | 3.97838 | 15.89591 |
| **предикатив** | 9.68289 | 32.32831 |
| **числівник** | 2.28924 | 37.08512 |

Критичне значення χ2 при ступені свободи *f = (K-1)(M-1) = 19* при довірчій імовірності 95% дорівнює **30,1**.

Отже, вибірки статистично однорідні відносно сполучника, прийменника, прислівника, вигука, компаративу і дієприслівника, адже їх критерій однорідності χ2 менше даного критичного значення. Це означає, що для цих частин мов нульова гіпотеза приймається.

Для іменника, дієслова, прикметника, займенника, частки, предикативу і числівника нульова гіпотеза відкидається, бо адже їх критерій однорідності χ2 більше даного критичного значення.

Критичне значення критерію Стьюдента при ступені свободи *f = n+m-2 = 38* при довірчій імовірності 95% дорівнює **2,04** (берем найбільш наближене до 38, що його не перевищує – 30).

За цим статистичним показником вибірки статистично однорідні відносно займенника, адже для нього критерій Стьюдента менше даного критичного значення. Це означає, що для цієї частини мови нульова гіпотеза приймається.

Для всіх інших частин мови нульова гіпотеза відкидається, бо адже їх значення критерію Стьюдента більше даного критичного значення.

# ВИСНОВКИ

В ході роботи було автоматично укладено частотні словники словоформ, лем і частин мови за двома вибірками. Вибірки було укладено за зональним принципом. Кожна складається з 20 тисяч слововживань, і в ході роботи їх було поділено на 20 підвибірок.

Для дослідження було взято науковий історичний текст та художній.

Знайдені статистичні характеристики в ході виконання другої частини роботи дозволяють зробити такі висновки.

1. В вибірці, що складається з наукового історичного тексту частіше зустрічаються іменник, прикметник та прийменник.
2. В вибірці з художнього тексту частіше зустрічаються всі інші виявлено частини мови. Числівник зустрічається частіше в художньому тексті через те, що після токенізації було залишено тільки слова, а в історичних текстах дати зазвичай пишуться цифрами.
3. Сполучник з однаковою частотою вживається в обох вибірках.
4. При зіставленні даних з вибірок за довірчими інтервалами 99% можна зробити висновок, що нульова гіпотеза приймається лише відносно сполучника та вигука. Тобто, в цих частин мови розходження між середніми значеннями абсолютних частот незначне.
5. При зіставлені вибірок за критерієм однорідності χ2 нульова гіпотеза приймається відносно сполучника, прийменника, прислівника, вигука, компаративу і дієприслівника.
6. При зіставлені вибірок за критерієм Стьюдента вибірки однорідні відносно займенника.

Отже, більшість статистичних даних вказують на те, що вибірки належать до різних генеральних сукупностей.

В ході виконання роботи виникали проблеми через те, що модуль pymorphy2 не завжди правильно визначає частиномовну приналежність слів, а також через омонімію.

## Додаток 1. Код для формування частотних словників за першим текстом.

*#!/usr/bin/env python  
# coding: utf-8***import** sqlite3  
**import** re  
**import** pymorphy2  
  
  
*#Створюємо базу даних і таблицю частотності словоформ*conn = sqlite3.connect(**'frequency\_dict.db'**)  
cursor = conn.cursor()  
  
cursor.execute(**'''CREATE TABLE IF NOT EXISTS част\_словоформ  
 (словоформа TEXT PRIMARY KEY NOT NULL,  
 gen\_freq INTEGER,  
 підв\_1 INTEGER,  
 підв\_2 INTEGER,  
 підв\_3 INTEGER,  
 підв\_4 INTEGER,  
 підв\_5 INTEGER,  
 підв\_6 INTEGER,  
 підв\_7 INTEGER,  
 підв\_8 INTEGER,  
 підв\_9 INTEGER,  
 підв\_10 INTEGER,  
 підв\_11 INTEGER,  
 підв\_12 INTEGER,  
 підв\_13 INTEGER,  
 підв\_14 INTEGER,  
 підв\_15 INTEGER,  
 підв\_16 INTEGER,  
 підв\_17 INTEGER,  
 підв\_18 INTEGER,  
 підв\_19 INTEGER,  
 підв\_20 INTEGER)  
'''**)  
  
cursor.execute(**'''CREATE TABLE IF NOT EXISTS част\_частин\_мови  
 (частина\_мови TEXT PRIMARY KEY NOT NULL,  
 gen\_freq INTEGER,  
 x\_aver INTEGER,  
 relative\_freq INTEGER,  
 підв\_1 INTEGER,  
 підв\_2 INTEGER,  
 підв\_3 INTEGER,  
 підв\_4 INTEGER,  
 підв\_5 INTEGER,  
 підв\_6 INTEGER,  
 підв\_7 INTEGER,  
 підв\_8 INTEGER,  
 підв\_9 INTEGER,  
 підв\_10 INTEGER,  
 підв\_11 INTEGER,  
 підв\_12 INTEGER,  
 підв\_13 INTEGER,  
 підв\_14 INTEGER,  
 підв\_15 INTEGER,  
 підв\_16 INTEGER,  
 підв\_17 INTEGER,  
 підв\_18 INTEGER,  
 підв\_19 INTEGER,  
 підв\_20 INTEGER)  
'''**)  
  
cursor.execute(**'''CREATE TABLE IF NOT EXISTS част\_лем  
 (лема TEXT PRIMARY KEY NOT NULL,  
 gen\_freq INTEGER,  
 підв\_1 INTEGER,  
 підв\_2 INTEGER,  
 підв\_3 INTEGER,  
 підв\_4 INTEGER,  
 підв\_5 INTEGER,  
 підв\_6 INTEGER,  
 підв\_7 INTEGER,  
 підв\_8 INTEGER,  
 підв\_9 INTEGER,  
 підв\_10 INTEGER,  
 підв\_11 INTEGER,  
 підв\_12 INTEGER,  
 підв\_13 INTEGER,  
 підв\_14 INTEGER,  
 підв\_15 INTEGER,  
 підв\_16 INTEGER,  
 підв\_17 INTEGER,  
 підв\_18 INTEGER,  
 підв\_19 INTEGER,  
 підв\_20 INTEGER)  
'''**)  
  
**with** open(**"вибірка 1.txt"**, encoding = **"utf-8"**) **as** data\_1:  
 text\_1 = data\_1.read().lower()  
 *#Робимо токенізацію*without\_punc\_marks = []  
edited = re.sub(**'\.'**, **''**, text\_1)  
edited = re.sub(**','**, **''**, edited)  
edited = re.sub(**'!'**, **''**, edited)  
edited = re.sub(**'\?'**, **''**, edited)  
edited = re.sub(**':'**, **''**, edited)  
edited = re.sub(**';'**, **''**, edited)  
edited = re.sub(**'\.\.'**, **''**, edited)  
edited = re.sub(**'\('**, **''**, edited)  
edited = re.sub(**'\)'**, **''**, edited)  
edited = re.sub(**'[a-zA-Z]+'**, **''**, edited)  
edited = re.sub(**'"'**, **''**, edited)  
edited = re.sub(**"'"**, **''**, edited)  
edited = re.sub(**'\['**, **''**, edited)  
edited = re.sub(**'\]'**, **''**, edited)  
edited = re.sub(**'\d'**, **''**, edited)  
edited = re.sub(**'%'**, **''**, edited)  
edited = re.sub(**'\\\\'**, **''**, edited)  
edited = re.sub(**'\n'**, **' '**, edited)  
edited = re.sub(**"'\]"**, **''**, edited)  
edited = re.sub(**"\['"**, **''**, edited)  
edited = re.sub(**'\s-\s'**, **' '**, edited)  
without\_punc\_marks.append(edited)

splitted\_1 = str(without\_punc\_marks).split(**' '**) *#Видаляємо пусті слова***try**:  
 **for** i **in** splitted\_1:  
 **if** i == **''**:  
 splitted\_1.remove(i)  
**except**:  
 print(**'Пустих слів немає'**)  
  
*#Відраховуємо потрібну кількість слововживань (20000)*count = 0  
edited\_list = []  
**for** word **in** splitted\_1:  
 **if** count == 20000:  
 **break** count += 1  
 edited\_list.append(word)  
sample\_1\_dict = {}  
sample\_1\_list = []  
  
*#Ділимо вибірку на підвибірки*count = 0  
x = 0  
**for** word **in** edited\_list:  
 **if** count == 1000:  
 **break** count += 1  
 sample\_1\_list.append(word)  
 **if** word **in** sample\_1\_dict:  
 sample\_1\_dict[word][1] += 1  
 **else**:  
 sample\_1\_dict[word] = [word]  
 sample\_1\_dict[word].append(1)  
sample\_2\_list = []  
count = 0  
**for** word **in** edited\_list:  
 count +=1  
  
 **if** 1000 < count < 2001:  
 sample\_2\_list.append(word)  
 **if** word **in** sample\_1\_dict:  
 **try**:  
 sample\_1\_dict[word][2] += 1  
 **except**:  
 sample\_1\_dict[word].append(1)  
 **else**:  
 sample\_1\_dict[word] = [word]  
 x = 0  
 **for** i **in** sample\_1\_dict[word]:  
 **while** x < 1:  
 sample\_1\_dict[word].append(0)  
 x += 1  
 sample\_1\_dict[word].append(1)  
**def** divide\_on\_samples(number):  
 **if** word **in** sample\_1\_dict:  
 **try**:  
 sample\_1\_dict[word][number] += 1  
 **except**:  
 sample\_len = len(sample\_1\_dict[word])  
 **if** sample\_len < number:  
 x = sample\_len  
 **while** x < number:  
 sample\_1\_dict[word].append(0)  
 x += 1  
 sample\_1\_dict[word].append(1)  
 **else**:  
 sample\_1\_dict[word].append(1)  
 **else**:  
 sample\_1\_dict[word] = [word]  
 x = 0  
 **while** x < number-1:  
 sample\_1\_dict[word].append(0)  
 x += 1  
 sample\_1\_dict[word].append(1)  
  
sample\_3\_list = []  
count = 0  
**for** word **in** edited\_list:  
 count +=1  
 **if** 2000 < count < 3001:  
 sample\_3\_list.append(word)  
 divide\_on\_samples(3)  
sample\_4\_list = []  
count = 0  
**for** word **in** edited\_list:  
 count +=1  
 **if** 3000 < count < 4001:  
 sample\_4\_list.append(word)  
 divide\_on\_samples(4)  
sample\_5\_list = []  
count = 0

**for** word **in** edited\_list:  
 count +=1  
 **if** 4000 < count < 5001:  
 sample\_5\_list.append(word)  
 divide\_on\_samples(5)  
sample\_6\_list = []  
count = 0  
**for** word **in** edited\_list:  
 count +=1  
 **if** 5000 < count < 6001:  
 sample\_6\_list.append(word)  
 divide\_on\_samples(6)  
sample\_7\_list = []  
count = 0  
**for** word **in** edited\_list:  
 count +=1  
 **if** 6000 < count < 7001:  
 sample\_7\_list.append(word)  
 divide\_on\_samples(7)  
sample\_8\_list = []  
count = 0  
**for** word **in** edited\_list:  
 count +=1  
 **if** 7000 < count < 8001:  
 sample\_8\_list.append(word)  
 divide\_on\_samples(8)  
sample\_9\_list = []  
count = 0  
**for** word **in** edited\_list:  
 count +=1  
 **if** 8000 < count < 9001:  
 sample\_9\_list.append(word)  
 divide\_on\_samples(9)  
sample\_10\_list = []  
count = 0  
**for** word **in** edited\_list:  
 count +=1  
 **if** 9000 < count < 10001:  
 sample\_10\_list.append(word)  
 divide\_on\_samples(10)  
sample\_11\_list = []  
count = 0  
**for** word **in** edited\_list:  
 count +=1  
 **if** 10000 < count < 11001:  
 sample\_11\_list.append(word)  
 divide\_on\_samples(11)  
sample\_12\_list = []  
count = 0  
**for** word **in** edited\_list:  
 count +=1  
 **if** 11000 < count < 12001:  
 sample\_12\_list.append(word)  
 divide\_on\_samples(12)  
sample\_13\_list = []  
count = 0  
**for** word **in** edited\_list:  
 count +=1  
 **if** 12000 < count < 13001:  
 sample\_13\_list.append(word)  
 divide\_on\_samples(13)sample\_14\_list = []  
count = 0  
**for** word **in** edited\_list:  
 count +=1  
 **if** 13000 < count < 14001:  
 sample\_14\_list.append(word)  
 divide\_on\_samples(14)  
sample\_15\_list = []  
count = 0  
**for** word **in** edited\_list:  
 count +=1  
 **if** 14000 < count < 15001:  
 sample\_15\_list.append(word)  
 divide\_on\_samples(15)  
sample\_16\_list = []  
count = 0  
**for** word **in** edited\_list:  
 count +=1  
 **if** 15000 < count < 16001:  
 sample\_16\_list.append(word)  
 divide\_on\_samples(16)  
sample\_17\_list = []  
count = 0  
**for** word **in**

edited\_list:  
 count +=1  
 **if** 16000 < count < 17001:  
 sample\_17\_list.append(word)  
 divide\_on\_samples(17)  
sample\_18\_list = []  
count = 0  
**for** word **in** edited\_list:  
 count +=1  
 **if** 17000 < count < 18001:  
 sample\_18\_list.append(word)  
 divide\_on\_samples(18)  
sample\_19\_list = []  
count = 0  
**for** word **in** edited\_list:  
 count +=1  
 **if** 18000 < count < 19001:  
 sample\_19\_list.append(word)  
 divide\_on\_samples(19)  
sample\_20\_list = []  
count = 0  
**for** word **in** edited\_list:  
 count +=1  
 **if** 19000 < count:  
 sample\_20\_list.append(word)  
 divide\_on\_samples(20)  
values1 = list(sample\_1\_dict.values())  
  
 *#Додаємо значення 0 до словоформ, які зустрілися лише в першій підвибірці*x = 2  
**for** i **in** values1:  
 **while** len(i) <21:  
 i.append(0)  
 x += 1  
  
  
 gen\_freq = sum(i[1:21]) *#додаємо до списку загальу кількість слововживань* i.insert(1, gen\_freq)  
values1\_ordered = []  
**for** i **in** values1:  
 values1\_ordered.append(i)  
**for** i **in** values1\_ordered:  
 count = 0  
*# while count in range(0, len(values1\_ordered)-1):  
# if values1\_ordered[count][1] < values1\_ordered[count+1][1]:  
# values1\_ordered[count], values1\_ordered[count+1] = values1\_ordered[count+1], values1\_ordered[count]  
# count += 1  
  
  
#for i in values1\_ordered:  
# cursor.execute("""INSERT INTO част\_словоформ  
# VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)""", i)  
#conn.commit()*morph = pymorphy2.MorphAnalyzer(lang=**'uk'**)  
part\_of\_speech = {}  
  
*# складаємо словник частин мови у першій підвибірці***for** i **in** sample\_1\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS**if** freq **in** part\_of\_speech:  
 part\_of\_speech[freq][1] += 1  
 **else**:  
 part\_of\_speech[freq] = [freq]  
 part\_of\_speech[freq].append(1)  
 *#print(i + ' ' + str(parsed1.tag.POS))  
#print(part\_of\_speech)***for** i **in** sample\_2\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS  
 *#print(freq)* **if** freq **in** part\_of\_speech:  
 **try**:  
 part\_of\_speech[freq][2] += 1  
 **except**:  
 part\_of\_speech[freq].append(1)  
 **else**:  
 part\_of\_speech[freq] = [freq]  
 x = 0  
 **for** o **in** part\_of\_speech[freq]:  
 **while** x < 1:  
 part\_of\_speech[freq].append(0)  
 x += 1  
 part\_of\_speech[freq].append(1)  
**def** count\_parts\_of\_speech(number):  
 **if** freq **in** part\_of\_speech:  
 **try**:  
 part\_of\_speech[freq][number] += 1  
 **except**:  
 freq\_len = len(part\_of\_speech[freq])  
 **if** freq\_len < number:  
 x = freq\_len  
 **while** x < number:  
 part\_of\_speech[freq].append(0)  
 x += 1  
 part\_of\_speech[freq].append(1)  
 **else**:  
 part\_of\_speech[freq].append(1)  
 **else**:  
 part\_of\_speech[freq] = [freq]  
 x = 0  
 **while** x < number-1:  
 part\_of\_speech[freq].append(0)  
 x += 1  
 part\_of\_speech[freq].append(1)  
  
**for** i **in** sample\_3\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS  
 *#print(freq)* count\_parts\_of\_speech(3)**for** i **in** sample\_4\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS  
 count\_parts\_of\_speech(4)**for** i **in** sample\_5\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS  
 *#print(freq)* count\_parts\_of\_speech(5)**for** i **in** sample\_6\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS  
 count\_parts\_of\_speech(6)**for** i **in** sample\_7\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS  
 count\_parts\_of\_speech(7)**for** i **in** sample\_8\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS  
 *#print(freq)* count\_parts\_of\_speech(8)**for** i **in** sample\_9\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS  
 count\_parts\_of\_speech(9)**for** i **in** sample\_10\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS  
 count\_parts\_of\_speech(10)**for** i **in** sample\_11\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS  
 **if** freq **in** part\_of\_speech:  
 count\_parts\_of\_speech(11)**for** i **in** sample\_12\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS  
 count\_parts\_of\_speech(12)**for** i **in** sample\_13\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS  
 count\_parts\_of\_speech(13)**for** i **in** sample\_14\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS  
 count\_parts\_of\_speech(14)**for** i **in** sample\_15\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS  
 count\_parts\_of\_speech(15)**for** i **in** sample\_16\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS  
 count\_parts\_of\_speech(16)**for** i **in** sample\_17\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS  
 count\_parts\_of\_speech(17)  
*#print(part\_of\_speech)***for** i **in** sample\_18\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS  
 count\_parts\_of\_speech(18)**for** i **in** sample\_14\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS  
 count\_parts\_of\_speech(19)**for** i **in** sample\_20\_list:  
 parsed1 = morph.parse(i)[0]  
 freq = parsed1.tag.POS  
 count\_parts\_of\_speech(20)values2 = list(part\_of\_speech.values())  
  
  
**for** i **in** values2:  
 **while** len(i) <21:  
 i.append(0)  
  
 gen\_freq = sum(i[1:21]) *#додаємо до списку загальу кількість слововживань* i.insert(1, gen\_freq)  
  
**for** i **in** values2:  
 **if** i[0] == **None**:  
 values2.remove(i)values2\_ordered = []  
**for** i **in** values2:  
 values2\_ordered.append(i)**for** i **in** values2\_ordered:  
 i.insert(2, i[1]/20)  
 i.insert(3, round(i[1]/20000, 3))**for** i **in** values2\_ordered:  
 count = 0  
 **while** count **in** range(0, len(values2\_ordered)-1):  
 **if** values2\_ordered[count][1] < values2\_ordered[count+1][1]:  
 values2\_ordered[count], values2\_ordered[count+1] = values2\_ordered[count+1], values2\_ordered[count]  
 count += 1 *#for i in values2\_ordered:  
# cursor.execute("""INSERT INTO част\_частин\_мови  
# VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)""", i)  
#conn.commit()*lemmas = {}  
**for** i **in** sample\_1\_list:  
 normal\_form = morph.parse(i)[0].normal\_form  
 *#print(normal\_form)* **if** normal\_form **in** lemmas:  
 lemmas[normal\_form][1] += 1  
 **else**:  
 lemmas[normal\_form] = [normal\_form]  
 lemmas[normal\_form].append(1)**for** i **in** sample\_2\_list:  
 normal\_form = morph.parse(i)[0].normal\_form  
 **if** normal\_form **in** lemmas:  
 **try**:  
 lemmas[normal\_form][2] += 1  
 **except**:  
 lemmas[normal\_form].append(1)  
 **else**:  
 lemmas[normal\_form] = [normal\_form]  
 x = 0  
 **for** o **in** lemmas[normal\_form]:  
 **while** x < 1:  
 lemmas[normal\_form].append(0)  
 x += 1  
 lemmas[normal\_form].append(1)**def** count\_lemmas(number):  
 **if** normal\_form **in** lemmas:  
 **try**:  
 lemmas[normal\_form][number] += 1  
 **except**:  
 lemmas\_len = len(lemmas[normal\_form])  
 **if** lemmas\_len < number:  
 x = lemmas\_len  
 **while** x < number:  
 lemmas[normal\_form].append(0)  
 x += 1  
 lemmas[normal\_form].append(1)  
 **else**:  
 lemmas[normal\_form].append(1)  
 **else**:  
 lemmas[normal\_form] = [normal\_form]  
 x = 0  
 **for** o **in** lemmas[normal\_form]:  
 **while** x < number-1:  
 lemmas[normal\_form].append(0)  
 x += 1  
 lemmas[normal\_form].append(1)  
  
**for** i **in** sample\_3\_list:  
 normal\_form = morph.parse(i)[0].normal\_form  
 count\_lemmas(3)**for** i **in** sample\_4\_list:  
 normal\_form = morph.parse(i)[0].normal\_form  
 count\_lemmas(4)

**for** i **in** sample\_5\_list:  
 normal\_form = morph.parse(i)[0].normal\_form  
 count\_lemmas(5)**for** i **in** sample\_6\_list:  
 normal\_form = morph.parse(i)[0].normal\_form  
 count\_lemmas(6)**for** i **in** sample\_7\_list:  
 normal\_form = morph.parse(i)[0].normal\_form  
 count\_lemmas(7)**for** i **in** sample\_8\_list:  
 normal\_form = morph.parse(i)[0].normal\_form  
 count\_lemmas(8)**for** i **in** sample\_9\_list:  
 normal\_form =

morph.parse(i)[0].normal\_form  
 count\_lemmas(9)**for** i **in** sample\_10\_list:  
 normal\_form = morph.parse(i)[0].normal\_form  
 count\_lemmas(10)**for** i **in** sample\_11\_list:  
 normal\_form = morph.parse(i)[0].normal\_form  
 count\_lemmas(11)**for** i **in** sample\_12\_list:  
 normal\_form = morph.parse(i)[0].normal\_form  
 count\_lemmas(12)**for** i **in** sample\_13\_list:  
 normal\_form = morph.parse(i)[0].normal\_form  
 count\_lemmas(13)**for** i **in** sample\_14\_list:  
 normal\_form = morph.parse(i)[0].normal\_form  
 count\_lemmas(14)**for** i **in** sample\_15\_list:  
 normal\_form = morph.parse(i)[0].normal\_form  
 count\_lemmas(15)**for** i **in** sample\_16\_list:  
 normal\_form = morph.parse(i)[0].normal\_form  
 count\_lemmas(16)**for** i **in** sample\_17\_list:  
 normal\_form = morph.parse(i)[0].normal\_form  
 count\_lemmas(17)**for** i **in** sample\_18\_list:  
 normal\_form = morph.parse(i)[0].normal\_form  
 count\_lemmas(18)**for** i **in** sample\_19\_list:  
 normal\_form = morph.parse(i)[0].normal\_form  
 count\_lemmas(19)**for** i **in** sample\_20\_list:  
 normal\_form = morph.parse(i)[0].normal\_form  
 count\_lemmas(20)values3 = list(lemmas.values())  
**for** i **in** values3:  
 **while** len(i) <21:  
 i.append(0)  
  
 gen\_freq = sum(i[1:21]) *#додаємо до списку загальу кількість слововживань* i.insert(1, gen\_freq)values3\_ordered = []  
**for** i **in** values3:  
 values3\_ordered.append(i)  
**for** i **in** values3\_ordered:  
 count = 0  
 **while** count **in** range(0, len(values3\_ordered)-1):  
 **if** values3\_ordered[count][1] < values3\_ordered[count+1][1]:  
 values3\_ordered[count], values3\_ordered[count+1] = values3\_ordered[count+1], values3\_ordered[count]  
 count += 1 *#for i in values3\_ordered:  
# cursor.execute("""INSERT INTO част\_лем  
# VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)""", i)  
#conn.commit()*conn.close()

# Додаток 2. Код для обчислення статистичних характеристик

*#!/usr/bin/env python  
# coding: utf-8***import** sqlite3  
**from** math **import** sqrt  
**from** tabulate **import** tabulate  
**import** matplotlib.pyplot **as** plt  
  
conn = sqlite3.connect(**'frequency\_dict.db'**)  
cursor = conn.cursor()  
  
**def** create\_tables():  
create\_tables()  
  
cursor.execute(**'''CREATE TABLE IF NOT EXISTS Стьюдент\_хі\_квадрат  
 (id INTEGER PRIMARY KEY NOT NULL,  
 part\_of\_sp TEXT,  
 student REAL,  
 xi\_squared REAL  
 )  
'''**)  
conn.commit()  
  
**def** count\_aver\_freq():  
 **global** aver\_freq\_x  
 **global** aver\_freq\_x\_ordered  
 **global** freq  
 **global** part\_of\_sp  
 aver\_freq\_x = {}rows = cursor.fetchall()  
 **for** row **in** rows:  
 freq = row[4:24]  
 part\_of\_sp = row[0]  
 **for** i **in** freq:  
 **if** i **in** aver\_freq\_x:  
 aver\_freq\_x[i][1] += 1  
 xini0 = aver\_freq\_x[i][0]\*aver\_freq\_x[i][1]  
 aver\_freq\_x[i][2] = xini0  
 **else**:  
 aver\_freq\_x[i] = [i]  
 aver\_freq\_x[i].append(1)  
 xini0 = aver\_freq\_x[i][0]\*aver\_freq\_x[i][1]  
 aver\_freq\_x[i].append(xini0)aver\_freq\_x = list(aver\_freq\_x.values())  
 aver\_freq\_x\_ordered = sorted(aver\_freq\_x, key=**lambda** x:x[0])  
 count=0  
 **while** count **in** range (0, len(aver\_freq\_x\_ordered)):  
 xini0= []  
 ni0 = []  
 **for** i **in** aver\_freq\_x\_ordered:  
 xini0.append(i[2])  
 ni0.append(i[1])  
 x\_aver0 = sum(xini0)/sum(ni0)  
 aver\_freq\_x\_ordered[count].append(x\_aver0)  
 aver\_freq\_x\_ordered[count].append((aver\_freq\_x\_ordered[count][0]-x\_aver0))  
 aver\_freq\_x\_ordered[count].append(((aver\_freq\_x\_ordered[count][0]-x\_aver0)\*\*2))  
 aver\_freq\_x\_ordered[count].append(((aver\_freq\_x\_ordered[count][0]-x\_aver0)\*\*2\*aver\_freq\_x\_ordered[count][1]))  
 count += 1  
 **def** statistics():  
 **global** xi, ni, xini, ni\_sum, x\_aver, xi\_minus\_x\_aver, xi\_minus\_x\_aver\_squared, xi\_minus\_x\_aver\_squared\_ni, sigma, \  
 sigma\_x\_aver, interval\_sigma, interval\_2\_sigma, interval\_3\_sigma, interval\_sigma\_x\_aver, interval\_sigma\_2\_x\_aver, \  
 interval\_sigma\_3\_x\_aver, coef\_of\_variation, coef\_of\_variation\_max, D, error  
 **global** my\_list  
 rows = cursor.fetchall()  
 xi = []  
 **for** row **in** rows:  
 xi.append(row[1])ni = []  
 **for** row **in** rows:  
 ni.append(row[2])  
 ni\_sum = sum(ni)xini = []  
 **for** row **in** rows:  
 xini.append(row[3])  
 xini\_sum = sum(xini)x\_aver = rows[0][4]xi\_minus\_x\_aver = []  
 **for** row **in** rows:  
 xi\_minus\_x\_aver.append(row[5])  
  
 xi\_minus\_x\_aver\_squared = []  
 **for** row **in** rows:  
 xi\_minus\_x\_aver\_squared.append(row[6])  
  
 xi\_minus\_x\_aver\_squared\_ni = []  
 **for** row **in** rows:  
 xi\_minus\_x\_aver\_squared\_ni.append(row[7])  
  
 sigma = sqrt(sum(xi\_minus\_x\_aver\_squared\_ni)/ni\_sum)sigma\_x\_aver = sigma/sqrt(ni\_sum)interval\_sigma = [round(x\_aver-sigma), round(x\_aver+sigma)]  
 interval\_2\_sigma = [round(x\_aver-2\*sigma), round(x\_aver+2\*sigma)]  
 interval\_3\_sigma = [round(x\_aver-3\*sigma), round(x\_aver+3\*sigma)]  
 interval\_sigma\_x\_aver = [round(x\_aver-sigma\_x\_aver), round(x\_aver+sigma\_x\_aver)]  
 interval\_sigma\_2\_x\_aver = [round(x\_aver-2\*sigma\_x\_aver), round(x\_aver+2\*sigma\_x\_aver)]  
 interval\_sigma\_3\_x\_aver = [round(x\_aver-3\*sigma\_x\_aver), round(x\_aver+3\*sigma\_x\_aver)]  
  
 coef\_of\_variation = round(sigma/x\_aver, 2)  
 coef\_of\_variation\_max = sqrt(ni\_sum-1)  
 D = 1-coef\_of\_variation/coef\_of\_variation\_max  
 error = round((1.96\*coef\_of\_variation)/sqrt(sum(ni)), 2)  
 my\_list = []  
 my\_list.append(sigma)  
 my\_list.append(sigma\_x\_aver)  
 my\_list.append(str(interval\_sigma))  
 my\_list.append(str(interval\_2\_sigma))  
 my\_list.append(str(interval\_3\_sigma))  
 my\_list.append(str(interval\_sigma\_x\_aver))  
 my\_list.append(str(interval\_sigma\_2\_x\_aver))  
 my\_list.append(str(interval\_sigma\_3\_x\_aver))  
 my\_list.append(coef\_of\_variation)  
 my\_list.append(coef\_of\_variation\_max)  
 my\_list.append(D)  
 my\_list.append(error)**def** print\_results():  
 table = {**'xi'**: xi,**'ni'**: ni, **'xi\*ni'**: xini, **'х̅'**: [x\_aver], **'xi - х̅'**: xi\_minus\_x\_aver,  
 **'(xi - х̅)^2'**: xi\_minus\_x\_aver\_squared, **'(xi - х̅)^2\*ni'**: xi\_minus\_x\_aver\_squared\_ni}  
 print(tabulate(table, headers=**'keys'**))  
  
 table = [[**'σ'**, **'σх̅'**, **'інт. із σ'**, **'інт. із 2σ'**,  
 **'інт. із 3σ'**, **'інт. σх̅'**, **'інт. 2σх̅'**,  
 **'інт. 3σх̅'**, **'V'**, **'Vmax'**, **'D'**, **'похибка'**], [round(sigma), round(sigma\_x\_aver),  
 interval\_sigma, interval\_2\_sigma, interval\_3\_sigma, interval\_sigma\_x\_aver, interval\_sigma\_2\_x\_aver,  
 interval\_sigma\_3\_x\_aver, coef\_of\_variation, coef\_of\_variation\_max, D, error]]  
 print(**'\n'**)  
 print(tabulate(table, headers=**'firstrow'**))  
  
  
 x\_aver\_minus\_sigma = round(interval\_sigma[0])  
 x\_aver\_plus\_sigma = round(interval\_sigma[1])*\* **global** ni\_in\_interval\_sigma\_sum  
 ni\_in\_interval\_sigma = []  
 **for** i **in** aver\_freq\_x\_ordered:  
 **if** i[0] **in** range(x\_aver\_minus\_sigma, x\_aver\_plus\_sigma+1):  
 ni\_in\_interval\_sigma.append(i[1])  
 ni\_in\_interval\_sigma\_sum = sum(ni\_in\_interval\_sigma)  
 percentage = round(ni\_in\_interval\_sigma\_sum\*100/ni\_sum, 1)  
  
 print(**'\n'** + **'В інтервалі x сер. ± сігма знаходиться ' "("** + str(sum(ni\_in\_interval\_sigma)) + **') - '** + str(percentage) + **'% абсолютниих частот'**)  
  
 **if** round((68.3 - percentage)\*100/68.3, 1) < 5:  
 print(**'Це вказує на хорошу узгодженіcть теоретично обчисленого й емпіричного результату'**)  
 **else**:  
 print(**'Це вказує на погану узгодженіcть теоретично обчисленого й емпіричного результату'**)  
  
 x\_aver\_minus\_2\_sigma = round(interval\_2\_sigma[0])  
 x\_aver\_plus\_2\_sigma = round(interval\_2\_sigma[1])  
 print(interval\_2\_sigma)  
 **global** ni\_in\_interval\_2\_sigma\_sum  
 ni\_in\_interval\_2\_sigma = []  
 **for** i **in** aver\_freq\_x\_ordered:  
 **if** i[0] **in** range(x\_aver\_minus\_2\_sigma, x\_aver\_plus\_2\_sigma+1):  
 ni\_in\_interval\_2\_sigma.append(i[1])  
 ni\_in\_interval\_2\_sigma\_sum = sum(ni\_in\_interval\_2\_sigma)  
 percentage = round(ni\_in\_interval\_2\_sigma\_sum\*100/ni\_sum, 1)  
 print(**'\n'** + **'В інтервалі x сер. ± 2\*сігма знаходиться '** + **"("** + str(sum(ni\_in\_interval\_2\_sigma)) + **') - '** + str(percentage) + **'% абсолютниих частот'**)  
  
 **if** round((95.5 - percentage)\*100/95.5, 1) < 5:  
 print(**'Це вказує на хорошу узгодженіcть теоретично обчисленого й емпіричного результату'**)  
 **else**:  
 print(**'Це вказує на погану узгодженіcть теоретично обчисленого й емпіричного результату'**)  
  
 x\_aver\_minus\_3\_sigma = round(interval\_3\_sigma[0])  
 x\_aver\_plus\_3\_sigma = round(interval\_3\_sigma[1])**global** ni\_in\_interval\_3\_sigma\_sum  
 ni\_in\_interval\_3\_sigma = []  
 **for** i **in** aver\_freq\_x\_ordered:  
 **if** i[0] **in** range(x\_aver\_minus\_3\_sigma, x\_aver\_plus\_3\_sigma+1):  
 ni\_in\_interval\_3\_sigma.append(i[1])  
 ni\_in\_interval\_3\_sigma\_sum = sum(ni\_in\_interval\_3\_sigma)  
 percentage = round(ni\_in\_interval\_3\_sigma\_sum\*100/ni\_sum, 1)  
 print(**'\n'** + **'В інтервалі x сер. ± 3\*сігма знаходиться '** + str(percentage) + **'% абсолютниих частот'**)  
  
 **if** round((99.7 - percentage)\*100/99.7, 1) < 5:  
 print(**'Це вказує на хорошу узгодженіcть теоретично обчисленого й емпіричного результату'**)  
 **else**:  
 print(**'Це вказує на погану узгодженіcть теоретично обчисленого й емпіричного результату'**)  
  
 x\_aver\_minus\_sigma\_x\_aver = round(interval\_sigma\_x\_aver[0])  
 x\_aver\_plus\_sigma\_x\_aver = round(interval\_sigma\_x\_aver[1])  
 print(**'\n'** + **'''З імовірністю 68.3% ми можемо стверджувати, що в даній генеральній сукупності середня частота даної частини мови коливатиметься в межах '''** + **'від '**+ str(x\_aver\_minus\_sigma\_x\_aver) + **' до '** + str(x\_aver\_plus\_sigma\_x\_aver))  
  
 x\_aver\_minus\_2\_sigma\_x\_aver = round(interval\_sigma\_2\_x\_aver[0])  
 x\_aver\_plus\_2\_sigma\_x\_aver = round(interval\_sigma\_2\_x\_aver[1])  
 print(**'\n'** + **'''З імовірністю 95.5% ми можемо стверджувати, що в даній генеральній сукупності середня частота даної частини мови коливатиметься в межах '''** + **'від '**+ str(x\_aver\_minus\_2\_sigma\_x\_aver) + **' до '** + str(x\_aver\_plus\_2\_sigma\_x\_aver))  
  
 x\_aver\_minus\_3\_sigma\_x\_aver = round(interval\_sigma\_3\_x\_aver[0])  
 x\_aver\_plus\_3\_sigma\_x\_aver = round(interval\_sigma\_3\_x\_aver[1])  
 print(**'\n'** + **'''З імовірністю 99.7% ми можемо стверджувати, що в даній генеральній сукупності середня частота даної частини мови коливатиметься в межах '''** + **'від '**+ str(x\_aver\_minus\_3\_sigma\_x\_aver) + **' до '** + str(x\_aver\_plus\_3\_sigma\_x\_aver))  
  
**def** interval\_row(K):  
 **global** interval\_row\_list *# список з межами інтервалів і серединами* **global** ni\_in\_intervals *# список з кількостями хі в кожному інтервалі* R = max(xi) - min(xi)h = R/K  
 interval\_row\_list = []  
 ni\_in\_intervals = []  
 xini\_intervals = []  
 **for** i **in** range(0, K):  
 interval\_row\_list.append([min(xi)+i\*h, round((min(xi)+i\*h)+h, 1), round((min(xi)+i\*h)+h/2, 1)])  
 ni\_for\_one\_interval = []  
 **for** o **in** aver\_freq\_x\_ordered:  
 **if** min(xi)+i\*h <= o[0] <= (min(xi)+i\*h)+h:ni\_for\_one\_interval.append(o[1])**else**:  
 **continue** ni\_in\_intervals.append(ni\_for\_one\_interval)  
 xini\_intervals.append([round((min(xi)+i\*h)+h/2, 1), sum(ni\_for\_one\_interval)])  
 ni\_interval\_sum = []  
 xini\_intervals\_sum = []  
 **for** i **in** xini\_intervals:  
 ni\_interval\_sum.append(i[1])  
 xini\_intervals\_sum.append(i[0]\*i[1])  
 ni\_interval\_sum = sum(ni\_interval\_sum)  
 xini\_intervals\_sum = sum(xini\_intervals\_sum)  
 x\_aver\_interval = xini\_intervals\_sum/ni\_interval\_sum  
 print(x\_aver\_interval)  
  
**def** student(x, y, s\_1\_x, s\_2\_x):  
 **global** t  
 t = abs(x-y)/sqrt(s\_1\_x+s\_2\_x)  
 print(**'Критерій Стьюдента: '** + str(t))  
  
**def** X\_squared\_count(M1, M2, K):  
 **global** X\_squared  
 X\_squared = []  
 **for** i **in** range(0, ni\_sum):  
 X\_squared.append(M1[i]\*\*2/(sum(M1)\*sum(K[i])))  
 X\_squared.append(M2[i]\*\*2/(sum(M2)\*sum(K[i])))  
 N = sum(M1)+sum(M2)  
 X\_squared = N\*((sum(X\_squared))-1)  
 print(**'xi квадрат '** + str(X\_squared))  
  
  
  
*#СТВОРЮЄМО ВАРІАЦІЙНИЙ РЯД ДЛЯ ІМЕННИКА*cursor.execute(**"""select \* from част\_частин\_мови  
 where частина\_мови = 'NOUN'"""**)  
count\_aver\_freq()**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO іменник\_середня\_частота (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер,  
 квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
M1 = freq  
  
*# ДОДАЄМО ДО ТАБЛИЦІ ЗНАЧЕННЯ Х СЕРЕДНЬОГО, ТА ЗНАЧЕННЯ, НЕОБХІДНІ ДЛЯ ПОДАЛЬШИХ РОЗРАХУНКІВ*cursor.execute(**"select \* from іменник\_середня\_частота"**)  
statistics()  
  
s\_1\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
x = x\_avercursor.execute(**""" INSERT INTO іменник\_статистичні\_дані (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
  
  
print(**'СТАТИСТИЧНІ ДАНІ ДЛЯ ІМЕННИКА, ВИБІРКА 1'** + **'\n'**)  
print\_results()  
*#СТВОРЮЄМО ПОЛІГОН ЧАСТОТ ІМЕННИКА  
#plt.plot(xi, ni, label = '1 вибірка', marker = 'o', markerfacecolor = 'blue', markersize = 4)*interval\_row(5)  
xi\_interval = []  
ni\_interval = []  
**for** i **in** interval\_row\_list:  
 xi\_interval.append(i[2])  
  
**for** i **in** ni\_in\_intervals:  
 ni\_interval.append(sum(i))  
  
*#plt.plot(xi\_interval, ni\_interval, label = '1 вибірка, за інтервальним рядом', marker = 'o', markersize = 4)*cursor.execute(**"""select \* from част\_частин\_мови\_2  
 where частина\_мови = 'NOUN'"""**)  
count\_aver\_freq()  
  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO іменник\_середня\_частота\_2 (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
M2 = freq  
K = list(zip(M1, M2))  
  
*# ДОДАЄМО ДО ТАБЛИЦІ ЗНАЧЕННЯ Х СЕРЕДНЬОГО, ТА ЗНАЧЕННЯ, НЕОБХІДНІ ДЛЯ ПОДАЛЬШИХ РОЗРАХУНКІВ*cursor.execute(**"select \* from іменник\_середня\_частота\_2"**)  
statistics()  
  
s\_2\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
y = x\_aver  
cursor.execute(**""" INSERT INTO іменник\_статистичні\_дані\_2 (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
  
  
print(**'СТАТИСТИЧНІ ДАНІ ДЛЯ ІМЕННИКА, ВИБІРКА 2'** + **'\n'**)  
print\_results()  
  
interval\_row(11)  
xi\_interval = []  
ni\_interval = []  
**for** i **in** interval\_row\_list:  
 xi\_interval.append(i[2])  
  
**for** i **in** ni\_in\_intervals:  
 ni\_interval.append(sum(i))  
  
*#plt.plot(xi\_interval, ni\_interval, label = '2 вибірка, за інтервальним рядом', marker = 'o', markersize = 4)  
#plt.plot(xi, ni, label = '2 вибірка', marker = 'o', markersize = 4)*plt.xlabel(**'xi'**)  
plt.ylabel(**'ni'**)  
plt.title(**'полігон частот іменника'**)  
plt.legend()  
*#plt.show()*student(x, y, s\_1\_x, s\_2\_x)  
X\_squared\_count(M1, M2, K)  
part\_of\_sp = part\_of\_sp  
  
cursor.execute(**"""INSERT INTO Стьюдент\_хі\_квадрат (part\_of\_sp, student, xi\_squared)  
 VALUES (?, ?, ?)"""**, (str(part\_of\_sp), t, X\_squared))  
conn.commit()  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови  
 where частина\_мови = 'VERB'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO дієслово\_середня\_частота (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_1\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
x = x\_aver  
cursor.execute(**"select \* from дієслово\_середня\_частота"**)  
statistics()  
cursor.execute(**""" INSERT INTO дієслово\_статистичні\_дані (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M1 = freq  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ ДІЄСЛОВА, ВИБІРКА 1'** + **'\n'**)  
print\_results()  
*#СТВОРЮЄМО ПОЛІГОН ЧАСТОТ ДІЄСЛОВА  
#plt.plot(xi, ni, label = '1 вибірка', marker = 'o', markerfacecolor = 'blue', markersize = 4)*interval\_row(9)  
xi\_interval = []  
ni\_interval = []  
**for** i **in** interval\_row\_list:  
 xi\_interval.append(i[2])  
  
**for** i **in** ni\_in\_intervals:  
 ni\_interval.append(sum(i))  
  
plt.plot(xi\_interval, ni\_interval, label = **'1 вибірка, за інтервальним рядом'**, marker = **'o'**, markersize = 4)  
  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови\_2  
 where частина\_мови = 'VERB'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO дієслово\_середня\_частота\_2 (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_2\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
y = x\_aver  
cursor.execute(**"select \* from дієслово\_середня\_частота\_2"**)  
statistics()  
cursor.execute(**""" INSERT INTO дієслово\_статистичні\_дані\_2 (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M2 = freq  
K = list(zip(M1, M2))  
  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ ДІЄСЛОВА, ВИБІРКА 2'** + **'\n'**)  
print\_results()  
*#plt.plot(xi, ni, label = '2 вибірка', marker = 'o', markersize = 4)*interval\_row(7)  
xi\_interval = []  
ni\_interval = []  
**for** i **in** interval\_row\_list:  
 xi\_interval.append(i[2])  
  
**for** i **in** ni\_in\_intervals:  
 ni\_interval.append(sum(i))  
  
plt.plot(xi\_interval, ni\_interval, label = **'2 вибірка, за інтервальним рядом'**, marker = **'o'**, markersize = 4)  
  
plt.xlabel(**'xi'**)  
plt.ylabel(**'ni'**)  
plt.title(**'полігон частот дієслова'**)  
plt.legend()  
plt.show()  
  
student(x, y, s\_1\_x, s\_2\_x)  
X\_squared\_count(M1, M2, K)  
part\_of\_sp = part\_of\_sp  
cursor.execute(**"""INSERT INTO Стьюдент\_хі\_квадрат (part\_of\_sp, student, xi\_squared)  
 VALUES (?, ?, ?)"""**, (str(part\_of\_sp), t, X\_squared))  
conn.commit()  
  
  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови  
 where частина\_мови = 'ADJF'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO прикметник\_середня\_частота (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_1\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
x = x\_aver  
cursor.execute(**"select \* from прикметник\_середня\_частота"**)  
statistics()  
cursor.execute(**""" INSERT INTO прикметник\_статистичні\_дані (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M1 = freq  
  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ ПРИКМЕТНИКА, ВИБІРКА 1'** + **'\n'**)  
print\_results()  
*#plt.plot(xi, ni, label = '1 вибірка', marker = 'o', markerfacecolor = 'blue', markersize = 4)*interval\_row(7)  
xi\_interval = []  
ni\_interval = []  
**for** i **in** interval\_row\_list:  
 xi\_interval.append(i[2])  
  
**for** i **in** ni\_in\_intervals:  
 ni\_interval.append(sum(i))  
  
plt.plot(xi\_interval, ni\_interval, label = **'1 вибірка, за інтервальним рядом'**, marker = **'o'**, markersize = 4)  
  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови\_2  
 where частина\_мови = 'ADJF'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO прикметник\_середня\_частота\_2 (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_2\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
y = x\_aver  
cursor.execute(**"select \* from прикметник\_середня\_частота\_2"**)  
statistics()  
cursor.execute(**""" INSERT INTO прикметник\_статистичні\_дані\_2 (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M2 = freq  
K = list(zip(M1, M2))  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ ПРИКМЕТНИКА, ВИБІРКА 2'** + **'\n'**)  
print\_results()  
*#plt.plot(xi, ni, label = '2 вибірка', marker = 'o', markersize = 4)*interval\_row(5)  
xi\_interval = []  
ni\_interval = []  
**for** i **in** interval\_row\_list:  
 xi\_interval.append(i[2])  
  
**for** i **in** ni\_in\_intervals:  
 ni\_interval.append(sum(i))  
  
plt.plot(xi\_interval, ni\_interval, label = **'2 вибірка, за інтервальним рядом'**, marker = **'o'**, markersize = 4)  
  
plt.xlabel(**'xi'**)  
plt.ylabel(**'ni'**)  
plt.title(**'полігон частот прикметника'**)  
plt.legend()  
plt.show()  
  
student(x, y, s\_1\_x, s\_2\_x)  
X\_squared\_count(M1, M2, K)  
part\_of\_sp = part\_of\_sp  
cursor.execute(**"""INSERT INTO Стьюдент\_хі\_квадрат (part\_of\_sp, student, xi\_squared)  
 VALUES (?, ?, ?)"""**, (str(part\_of\_sp), t, X\_squared))  
conn.commit()  
  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови  
 where частина\_мови = 'CONJ'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO сполучник\_середня\_частота (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
*#print(aver\_freq\_conj\_ordered)*s\_1\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
x = x\_aver  
cursor.execute(**"select \* from сполучник\_середня\_частота"**)  
statistics()  
cursor.execute(**""" INSERT INTO сполучник\_статистичні\_дані (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M1 = freq  
  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ СПОЛУЧНИКА, ВИБІРКА 1'** + **'\n'**)  
print\_results()  
*#plt.plot(xi, ni, label = '1 вибірка', marker = 'o', markerfacecolor = 'blue', markersize = 4)*interval\_row(5)  
xi\_interval = []  
ni\_interval = []  
**for** i **in** interval\_row\_list:  
 xi\_interval.append(i[2])  
  
**for** i **in** ni\_in\_intervals:  
 ni\_interval.append(sum(i))  
  
plt.plot(xi\_interval, ni\_interval, label = **'1 вибірка, за інтервальним рядом'**, marker = **'o'**, markersize = 4)  
*#plt.legend()  
#plt.show()*cursor.execute(**"""select \* from част\_частин\_мови\_2  
 where частина\_мови = 'CONJ'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO сполучник\_середня\_частота\_2 (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_2\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
y = x\_aver  
cursor.execute(**"select \* from сполучник\_середня\_частота\_2"**)  
statistics()  
cursor.execute(**""" INSERT INTO сполучник\_статистичні\_дані\_2 (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M2 = freq  
K = list(zip(M1, M2))  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ СПОЛУЧНИКА, ВИБІРКА 2'** + **'\n'**)  
print\_results()  
  
interval\_row(7)  
xi\_interval = []  
ni\_interval = []  
**for** i **in** interval\_row\_list:  
 xi\_interval.append(i[2])  
  
**for** i **in** ni\_in\_intervals:  
 ni\_interval.append(sum(i))  
  
plt.plot(xi\_interval, ni\_interval, label = **'2 вибірка, за інтервальним рядом'**, marker = **'o'**, markersize = 4)  
  
*#plt.plot(xi, ni, label = '2 вибірка', marker = 'o', markersize = 4)*plt.xlabel(**'xi'**)  
plt.ylabel(**'ni'**)  
plt.title(**'полігон частот сполучника'**)  
plt.legend()  
plt.show()  
  
student(x, y, s\_1\_x, s\_2\_x)  
X\_squared\_count(M1, M2, K)  
part\_of\_sp = part\_of\_sp  
cursor.execute(**"""INSERT INTO Стьюдент\_хі\_квадрат (part\_of\_sp, student, xi\_squared)  
 VALUES (?, ?, ?)"""**, (str(part\_of\_sp), t, X\_squared))  
conn.commit()  
  
  
  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови  
 where частина\_мови = 'NPRO'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO займенниковий\_іменник\_середня\_частота (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
*#print(aver\_freq\_npro\_ordered)*s\_1\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
x = x\_aver  
cursor.execute(**"select \* from займенниковий\_іменник\_середня\_частота"**)  
statistics()  
cursor.execute(**""" INSERT INTO займенниковий\_іменник\_статистичні\_дані (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
M1 = freq  
  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ ЗАЙМЕННИКОВОГО ІМЕННИКА, ВИБІРКА 1'** + **'\n'**)  
print\_results()  
*#plt.plot(xi, ni, label = '1 вибірка', marker = 'o', markerfacecolor = 'blue', markersize = 4)*interval\_row(9)  
xi\_interval = []  
ni\_interval = []  
**for** i **in** interval\_row\_list:  
 xi\_interval.append(i[2])  
  
**for** i **in** ni\_in\_intervals:  
 ni\_interval.append(sum(i))  
  
plt.plot(xi\_interval, ni\_interval, label = **'1 вибірка, за інтервальним рядом'**, marker = **'o'**, markersize = 4)  
  
  
cursor.execute(**"""select \* from част\_частин\_мови\_2  
 where частина\_мови = 'NPRO'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO займенниковий\_іменник\_середня\_частота\_2 (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_2\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
y = x\_aver  
cursor.execute(**"select \* from займенниковий\_іменник\_середня\_частота\_2"**)  
statistics()  
cursor.execute(**""" INSERT INTO займенниковий\_іменник\_статистичні\_дані\_2 (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M2 = freq  
K = list(zip(M1, M2))  
  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ ЗАЙМЕННИКОВОГО ІМЕННИКА, ВИБІРКА 2'** + **'\n'**)  
print\_results()  
*#plt.plot(xi, ni, label = '2 вибірка', marker = 'o', markersize = 4)*interval\_row(5)  
xi\_interval = []  
ni\_interval = []  
**for** i **in** interval\_row\_list:  
 xi\_interval.append(i[2])  
  
**for** i **in** ni\_in\_intervals:  
 ni\_interval.append(sum(i))  
  
plt.plot(xi\_interval, ni\_interval, label = **'2 вибірка, за інтервальним рядом'**, marker = **'o'**, markersize = 4)  
  
plt.xlabel(**'xi'**)  
plt.ylabel(**'ni'**)  
plt.title(**'полігон частот займенникового іменника'**)  
plt.legend()  
plt.show()  
  
student(x, y, s\_1\_x, s\_2\_x)  
X\_squared\_count(M1, M2, K)  
part\_of\_sp = part\_of\_sp  
cursor.execute(**"""INSERT INTO Стьюдент\_хі\_квадрат (part\_of\_sp, student, xi\_squared)  
 VALUES (?, ?, ?)"""**, (str(part\_of\_sp), t, X\_squared))  
conn.commit()  
  
  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови  
 where частина\_мови = 'PREP'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO прийменник\_середня\_частота (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_1\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
x = x\_aver  
cursor.execute(**"select \* from прийменник\_середня\_частота"**)  
statistics()  
cursor.execute(**""" INSERT INTO прийменник\_статистичні\_дані (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M1 = freq  
  
  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ ПРИЙМЕННИКА, ВИБІРКА 1'** + **'\n'**)  
print\_results()  
*#plt.plot(xi, ni, label = '1 вибірка', marker = 'o', markerfacecolor = 'blue', markersize = 4)*interval\_row(7)  
xi\_interval = []  
ni\_interval = []  
**for** i **in** interval\_row\_list:  
 xi\_interval.append(i[2])  
  
**for** i **in** ni\_in\_intervals:  
 ni\_interval.append(sum(i))  
  
plt.plot(xi\_interval, ni\_interval, label = **'1 вибірка, за інтервальним рядом'**, marker = **'o'**, markersize = 4)  
  
  
  
  
  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови\_2  
 where частина\_мови = 'PREP'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO прийменник\_середня\_частота\_2 (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_2\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
y = x\_aver  
cursor.execute(**"select \* from прийменник\_середня\_частота\_2"**)  
statistics()  
cursor.execute(**""" INSERT INTO прийменник\_статистичні\_дані\_2 (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M2 = freq  
K = list(zip(M1, M2))  
  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ ПРИЙМЕННИКА, ВИБІРКА 2'** + **'\n'**)  
print\_results()  
*#plt.plot(xi, ni, label = '2 вибірка', marker = 'o', markersize = 4)*interval\_row(7)  
xi\_interval = []  
ni\_interval = []  
**for** i **in** interval\_row\_list:  
 xi\_interval.append(i[2])  
  
**for** i **in** ni\_in\_intervals:  
 ni\_interval.append(sum(i))  
  
plt.plot(xi\_interval, ni\_interval, label = **'2 вибірка, за інтервальним рядом'**, marker = **'o'**, markersize = 4)  
  
  
plt.xlabel(**'xi'**)  
plt.ylabel(**'ni'**)  
plt.title(**'полігон частот прийменника'**)  
plt.legend()  
plt.show()  
  
student(x, y, s\_1\_x, s\_2\_x)  
X\_squared\_count(M1, M2, K)  
part\_of\_sp = part\_of\_sp  
cursor.execute(**"""INSERT INTO Стьюдент\_хі\_квадрат (part\_of\_sp, student, xi\_squared)  
 VALUES (?, ?, ?)"""**, (str(part\_of\_sp), t, X\_squared))  
conn.commit()  
  
  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови  
 where частина\_мови = 'PRCL'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO частка\_середня\_частота (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_1\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
x = x\_aver  
cursor.execute(**"select \* from частка\_середня\_частота"**)  
statistics()  
cursor.execute(**""" INSERT INTO частка\_статистичні\_дані (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M1 = freq  
  
  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ ЧАСТКИ, ВИБІРКА 1'** + **'\n'**)  
print\_results()  
*#plt.plot(xi, ni, label = '1 вибірка', marker = 'o', markerfacecolor = 'blue', markersize = 4)*interval\_row(7)  
xi\_interval = []  
ni\_interval = []  
**for** i **in** interval\_row\_list:  
 xi\_interval.append(i[2])  
  
**for** i **in** ni\_in\_intervals:  
 ni\_interval.append(sum(i))  
  
plt.plot(xi\_interval, ni\_interval, label = **'1 вибірка, за інтервальним рядом'**, marker = **'o'**, markersize = 4)  
  
  
  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови\_2  
 where частина\_мови = 'PRCL'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO частка\_середня\_частота\_2 (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_2\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
y = x\_aver  
cursor.execute(**"select \* from частка\_середня\_частота\_2"**)  
statistics()  
cursor.execute(**""" INSERT INTO частка\_статистичні\_дані\_2 (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M2 = freq  
K = list(zip(M1, M2))  
  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ ЧАСТКИ, ВИБІРКА 2'** + **'\n'**)  
print\_results()  
  
interval\_row(7)  
xi\_interval = []  
ni\_interval = []  
**for** i **in** interval\_row\_list:  
 xi\_interval.append(i[2])  
  
**for** i **in** ni\_in\_intervals:  
 ni\_interval.append(sum(i))  
  
plt.plot(xi\_interval, ni\_interval, label = **'2 вибірка, за інтервальним рядом'**, marker = **'o'**, markersize = 4)  
  
*#plt.plot(xi, ni, label = '2 вибірка', marker = 'o', markersize = 4)*plt.xlabel(**'xi'**)  
plt.ylabel(**'ni'**)  
plt.title(**'полігон частот частки'**)  
plt.legend()  
plt.show()  
  
student(x, y, s\_1\_x, s\_2\_x)  
X\_squared\_count(M1, M2, K)  
part\_of\_sp = part\_of\_sp  
cursor.execute(**"""INSERT INTO Стьюдент\_хі\_квадрат (part\_of\_sp, student, xi\_squared)  
 VALUES (?, ?, ?)"""**, (str(part\_of\_sp), t, X\_squared))  
conn.commit()  
  
  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови  
 where частина\_мови = 'ADVB'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO прислівник\_середня\_частота (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_1\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
x = x\_aver  
cursor.execute(**"select \* from прислівник\_середня\_частота"**)  
statistics()  
cursor.execute(**""" INSERT INTO прислівник\_статистичні\_дані (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M1 = freq  
  
  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ ПРИСЛІВНИКА, ВИБІРКА 1'** + **'\n'**)  
print\_results()  
*#plt.plot(xi, ni, label = '1 вибірка', marker = 'o', markerfacecolor = 'blue', markersize = 4)*interval\_row(7)  
xi\_interval = []  
ni\_interval = []  
**for** i **in** interval\_row\_list:  
 xi\_interval.append(i[2])  
  
**for** i **in** ni\_in\_intervals:  
 ni\_interval.append(sum(i))  
  
plt.plot(xi\_interval, ni\_interval, label = **'1 вибірка, за інтервальним рядом'**, marker = **'o'**, markersize = 4)  
  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови\_2  
 where частина\_мови = 'ADVB'"""**)  
count\_aver\_freq()  
*#print(aver\_freq\_x\_ordered)***for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO прислівник\_середня\_частота\_2 (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_2\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
y = x\_aver  
cursor.execute(**"select \* from прислівник\_середня\_частота\_2"**)  
statistics()  
cursor.execute(**""" INSERT INTO прислівник\_статистичні\_дані\_2 (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M2 = freq  
K = list(zip(M1, M2))  
  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ ПРИСЛІВНИКА, ВИБІРКА 2'** + **'\n'**)  
print\_results()  
*#plt.plot(xi, ni, label = '2 вибірка', marker = 'o', markersize = 4)*interval\_row(7)  
xi\_interval = []  
ni\_interval = []  
**for** i **in** interval\_row\_list:  
 xi\_interval.append(i[2])  
  
**for** i **in** ni\_in\_intervals:  
 ni\_interval.append(sum(i))  
  
plt.plot(xi\_interval, ni\_interval, label = **'2 вибірка, за інтервальним рядом'**, marker = **'o'**, markersize = 4)  
  
  
plt.xlabel(**'xi'**)  
plt.ylabel(**'ni'**)  
plt.title(**'полігон частот прислівника'**)  
plt.legend()  
plt.show()  
  
student(x, y, s\_1\_x, s\_2\_x)  
X\_squared\_count(M1, M2, K)  
part\_of\_sp = part\_of\_sp  
cursor.execute(**"""INSERT INTO Стьюдент\_хі\_квадрат (part\_of\_sp, student, xi\_squared)  
 VALUES (?, ?, ?)"""**, (str(part\_of\_sp), t, X\_squared))  
conn.commit()  
  
  
  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови  
 where частина\_мови = 'INTJ'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO вигук\_середня\_частота (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_1\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
x = x\_aver  
cursor.execute(**"select \* from вигук\_середня\_частота"**)  
statistics()  
cursor.execute(**""" INSERT INTO вигук\_статистичні\_дані (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M1 = freq  
  
  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ ВИГУКА, ВИБІРКА 1'** + **'\n'**)  
print\_results()  
*#plt.plot(xi, ni, label = '1 вибірка', marker = 'o', markerfacecolor = 'blue', markersize = 4)*interval\_row(5)  
xi\_interval = []  
ni\_interval = []  
**for** i **in** interval\_row\_list:  
 xi\_interval.append(i[2])  
  
**for** i **in** ni\_in\_intervals:  
 ni\_interval.append(sum(i))  
  
plt.plot(xi\_interval, ni\_interval, label = **'1 вибірка, за інтервальним рядом'**, marker = **'o'**, markersize = 4)  
  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови\_2  
 where частина\_мови = 'INTJ'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO вигук\_середня\_частота\_2 (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_2\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
y = x\_aver  
cursor.execute(**"select \* from вигук\_середня\_частота\_2"**)  
statistics()  
cursor.execute(**""" INSERT INTO вигук\_статистичні\_дані\_2 (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M2 = freq  
K = list(zip(M1, M2))  
  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ ВИГУКА, ВИБІРКА 2'** + **'\n'**)  
print\_results()  
*#plt.plot(xi, ni, label = '2 вибірка', marker = 'o', markersize = 4)*interval\_row(5)  
xi\_interval = []  
ni\_interval = []  
**for** i **in** interval\_row\_list:  
 xi\_interval.append(i[2])  
  
**for** i **in** ni\_in\_intervals:  
 ni\_interval.append(sum(i))  
  
plt.plot(xi\_interval, ni\_interval, label = **'2 вибірка, за інтервальним рядом'**, marker = **'o'**, markersize = 4)  
  
  
plt.xlabel(**'xi'**)  
plt.ylabel(**'ni'**)  
plt.title(**'полігон частот вигука'**)  
plt.legend()  
plt.show()  
  
student(x, y, s\_1\_x, s\_2\_x)  
X\_squared\_count(M1, M2, K)  
part\_of\_sp = part\_of\_sp  
cursor.execute(**"""INSERT INTO Стьюдент\_хі\_квадрат (part\_of\_sp, student, xi\_squared)  
 VALUES (?, ?, ?)"""**, (str(part\_of\_sp), t, X\_squared))  
conn.commit()  
  
  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови  
 where частина\_мови = 'COMP'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO компаратив\_середня\_частота (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_1\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
x = x\_aver  
cursor.execute(**"select \* from компаратив\_середня\_частота"**)  
statistics()  
cursor.execute(**""" INSERT INTO компаратив\_статистичні\_дані (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M1 = freq  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ КОМПАРАТИВУ, ВИБІРКА 1'** + **'\n'**)  
print\_results()  
plt.plot(xi, ni, label = **'1 вибірка'**, marker = **'o'**, markerfacecolor = **'blue'**, markersize = 4)  
  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови\_2  
 where частина\_мови = 'COMP'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO компаратив\_середня\_частота\_2 (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_2\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
y = x\_aver  
cursor.execute(**"select \* from компаратив\_середня\_частота\_2"**)  
statistics()  
cursor.execute(**""" INSERT INTO компаратив\_статистичні\_дані\_2 (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M2 = freq  
K = list(zip(M1, M2))  
  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ КОМПАРАТИВУ, ВИБІРКА 2'** + **'\n'**)  
print\_results()  
plt.plot(xi, ni, label = **'2 вибірка'**, marker = **'o'**, markersize = 4)  
plt.xlabel(**'xi'**)  
plt.ylabel(**'ni'**)  
plt.title(**'полігон частот компаративу'**)  
plt.legend()  
plt.show()  
  
student(x, y, s\_1\_x, s\_2\_x)  
X\_squared\_count(M1, M2, K)  
part\_of\_sp = part\_of\_sp  
cursor.execute(**"""INSERT INTO Стьюдент\_хі\_квадрат (part\_of\_sp, student, xi\_squared)  
 VALUES (?, ?, ?)"""**, (str(part\_of\_sp), t, X\_squared))  
conn.commit()  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови  
 where частина\_мови = 'GRND'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO дієприслівник\_середня\_частота (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_1\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
x = x\_aver  
cursor.execute(**"select \* from дієприслівник\_середня\_частота"**)  
statistics()  
cursor.execute(**""" INSERT INTO дієприслівник\_статистичні\_дані (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M1 = freq  
  
  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ ДІЄПРИСЛІВНИКА, ВИБІРКА 1'** + **'\n'**)  
print\_results()  
plt.plot(xi, ni, label = **'1 вибірка'**, marker = **'o'**, markerfacecolor = **'blue'**, markersize = 4)  
  
  
  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови\_2  
 where частина\_мови = 'GRND'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO дієприслівник\_середня\_частота\_2 (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_2\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
y = x\_aver  
cursor.execute(**"select \* from дієприслівник\_середня\_частота\_2"**)  
statistics()  
cursor.execute(**""" INSERT INTO дієприслівник\_статистичні\_дані\_2 (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M2 = freq  
K = list(zip(M1, M2))  
  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ ДІЄПРИСЛІВНИКА, ВИБІРКА 2'** + **'\n'**)  
print\_results()  
plt.plot(xi, ni, label = **'2 вибірка'**, marker = **'o'**, markersize = 4)  
plt.xlabel(**'xi'**)  
plt.ylabel(**'ni'**)  
plt.title(**'полігон частот дієприслівника'**)  
plt.legend()  
plt.show()  
  
student(x, y, s\_1\_x, s\_2\_x)  
X\_squared\_count(M1, M2, K)  
part\_of\_sp = part\_of\_sp  
cursor.execute(**"""INSERT INTO Стьюдент\_хі\_квадрат (part\_of\_sp, student, xi\_squared)  
 VALUES (?, ?, ?)"""**, (str(part\_of\_sp), t, X\_squared))  
conn.commit()  
  
  
  
  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови  
 where частина\_мови = 'PRED'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO предикатив\_середня\_частота (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_1\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
x = x\_aver  
cursor.execute(**"select \* from предикатив\_середня\_частота"**)  
statistics()  
cursor.execute(**""" INSERT INTO предикатив\_статистичні\_дані (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M1 = freq  
  
  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ ПРЕДИКАТИВА, ВИБІРКА 1'** + **'\n'**)  
print\_results()  
plt.plot(xi, ni, label = **'1 вибірка'**, marker = **'o'**, markerfacecolor = **'blue'**, markersize = 4)  
  
  
  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови\_2  
 where частина\_мови = 'PRED'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO предикатив\_середня\_частота\_2 (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_2\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
y = x\_aver  
cursor.execute(**"select \* from предикатив\_середня\_частота\_2"**)  
statistics()  
cursor.execute(**""" INSERT INTO предикатив\_статистичні\_дані\_2 (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M2 = freq  
K = list(zip(M1, M2))  
  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ ПРЕДИКАТИВА, ВИБІРКА 2'** + **'\n'**)  
print\_results()  
plt.plot(xi, ni, label = **'2 вибірка'**, marker = **'o'**, markersize = 4)  
plt.xlabel(**'xi'**)  
plt.ylabel(**'ni'**)  
plt.title(**'полігон частот предикативу'**)  
plt.legend()  
plt.show()  
  
student(x, y, s\_1\_x, s\_2\_x)  
X\_squared\_count(M1, M2, K)  
part\_of\_sp = part\_of\_sp  
cursor.execute(**"""INSERT INTO Стьюдент\_хі\_квадрат (part\_of\_sp, student, xi\_squared)  
 VALUES (?, ?, ?)"""**, (str(part\_of\_sp), t, X\_squared))  
conn.commit()  
  
  
  
  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови  
 where частина\_мови = 'NUMR'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO числівник\_середня\_частота (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_1\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
x = x\_aver  
cursor.execute(**"select \* from числівник\_середня\_частота"**)  
statistics()  
cursor.execute(**""" INSERT INTO числівник\_статистичні\_дані (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M1 = freq  
  
  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ ЧИСЛІВНИКА, ВИБІРКА 1'** + **'\n'**)  
print\_results()  
plt.plot(xi, ni, label = **'1 вибірка'**, marker = **'o'**, markerfacecolor = **'blue'**, markersize = 4)  
  
  
  
  
  
cursor.execute(**"""select \* from част\_частин\_мови\_2  
 where частина\_мови = 'NUMR'"""**)  
count\_aver\_freq()  
**for** i **in** aver\_freq\_x\_ordered:  
 cursor.execute(**"""INSERT INTO числівник\_середня\_частота\_2 (xi, ni, xini, x\_сер, різниця\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_сер, квадрат\_різниці\_xi\_та\_x\_серni)  
 VALUES(?, ?, ?, ?, ?, ?, ?)"""**, i)  
conn.commit()  
  
s\_2\_x = sum(xi\_minus\_x\_aver\_squared\_ni)/(ni\_sum\*(ni\_sum-1))  
y = x\_aver  
cursor.execute(**"select \* from числівник\_середня\_частота\_2"**)  
statistics()  
cursor.execute(**""" INSERT INTO числівник\_статистичні\_дані\_2 (серед\_квадратич\_відхил, міра\_колив\_серед\_част,  
 x\_сер\_плюс\_мінус\_сигма, x\_сер\_плюс\_мінус\_2\_сигма, x\_сер\_плюс\_мінус\_3\_сигма, інтервал\_міри\_колив\_сигма,  
 інтервал\_міри\_колив\_2\_сигма, інтервал\_міри\_колив\_3\_сигма, V, Vmax, D, error)  
 VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"""**, my\_list)  
conn.commit()  
  
M2 = freq  
K = list(zip(M1, M2))  
  
print(**'\n'** + **'СТАТИСТИЧНІ ДАНІ ДЛЯ ЧИСЛІВНИКА, ВИБІРКА 2'** + **'\n'**)  
print\_results()  
plt.plot(xi, ni, label = **'2 вибірка'**, marker = **'o'**, markersize = 4)  
plt.xlabel(**'xi'**)  
plt.ylabel(**'ni'**)  
plt.title(**'полігон частот числівника'**)  
plt.legend()  
plt.show()  
  
student(x, y, s\_1\_x, s\_2\_x)  
X\_squared\_count(M1, M2, K)  
part\_of\_sp = part\_of\_sp  
cursor.execute(**"""INSERT INTO Стьюдент\_хі\_квадрат (part\_of\_sp, student, xi\_squared)  
 VALUES (?, ?, ?)"""**, (str(part\_of\_sp), t, X\_squared))  
conn.commit()  
  
conn.close()