

3. Файлы

Расскажем, как организовать ввод информации в файлы и вывод информации из файлов.

Чтобы читать данные из файла, либо писать данные в файл, файл надо открыть:

файловая_переменная = open(*имя_файла*, mode = *режим*, encoding = *кодировка*)

Режимы открытия файла: 'r' – открытие для чтения; 'w' – открытие для записи (файл создаётся заново; если файл был, старая версия удаляется); 'x' – открытие для записи (файл создаётся заново, если его не было; если файл был, возвращается ошибка); 't' – открытие файла как текстового; 'b' – открытие файла как двоичного; 'a' – открытие файла для записи в конец.

Кодировки: 'utf-8', 'utf-16' (кодировки Unicode), 'cp1251' (кодировка Windows русско-язычная), 'cp866' (кодировка DOS).

Чтобы прочесть все данные из файла *переменную*, используется следующий метод:

переменная = *файловая_переменная*.read()

Чтобы прочесть строку из файла в *переменную*, используется следующий метод:

переменная = *файловая_переменная*.readline()

Чтобы прочесть все строки из файла в *список* строк, используется следующий метод:

список = *файловая_переменная*.readlines()

Чтобы записать данные в файл, используется следующий метод:

файловая_переменная.write(*данные*)

Чтобы сохранить все изменения, сделанные в файле, используется следующий метод:

файловая_переменная.close()

Пример: создать два текстовых файла, заполнив их десятью случайными целыми числами из промежутка 0..9, а затем создать третий файл, заполнив его всеми попарными произведениями указанных чисел.

Листинг 1.

```
from random import *
a = open('a.txt', mode = 'w', encoding = 'utf-8')
b = open('b.txt', mode = 'w', encoding = 'utf-8')
for n in range(10):
    a.write(str(randint(0,10))+'\n')
    b.write(str(randint(0,10))+'\n')
a.close()
b.close()
a = open('a.txt', mode = 'r', encoding = 'utf-8')
b = open('b.txt', mode = 'r', encoding = 'utf-8')
c = open('c.txt', mode = 'w', encoding = 'utf-8')
ax = [int(line.rstrip()) for line in a.readlines()]
bx = [int(line.rstrip()) for line in b.readlines()]
cx = [[i * j for j in bx] for i in ax]
print(ax)
print(bx)
print(cx)
for i in range(len(ax)):
    for j in range(len(bx)):
        c.write(f"{cx[i][j]:5d}")
    c.write('\n')
a.close()
```

```
b.close()
c.close()
```

4. Применение python при исследовании процессов, протекающих в геологии

Моделирование процессов, заданных рекуррентно

В исследовательских задачах часто встает вопрос табулирования функций. Аргументом таких функций является время t , которое меняется от t_0 до t_n с некоторым шагом Δt (n – некоторое натуральное число, обозначающее общее количество шагов).

Размер шага всегда можно найти: $\Delta t = \frac{t_n - t_0}{n}$.

В таком случае для шага с произвольным номером i момент времени t_i вычисляется по формуле арифметической прогрессии: $t_i = t_0 + \Delta t \cdot i$.

Допустим, что функция $f(t_i)$, которая описывает некоторый процесс, вычисляется по рекуррентной (возвратной) формуле: $f(t_i) = F(f(t_{i-1}), t_{i-1})$, то есть, значение функции в момент времени t_i вычисляется через значение функции в момент времени t_{i-1} . В этом случае, вычислив значение функции в точке t_0 , можно получить значение функции в точке t_1 , после этого в t_2 и так далее.

Рассмотрим пример. Из реки в озеро вымываются некоторые осадочные вещества, причём выяснилось, что с каждым годом прирост этих веществ уменьшается вдвое. Протабулировать значения: прирост dv удельного количества намываемого вещества и общее удельное количество v вещества в озере. На начало исследований удельное количество вещества в озере составляло 59%, а удельный прирост за первый период составил 0.8%. Всего $n = 10$ периодов.

Имеем формулы:

$$\begin{aligned}t_i &= i, \\v &= v + dv, \\dv &= dv/2.\end{aligned}$$

В этом случае получаем программу:

Листинг 1.

```
t = 0
v = 59
dv = 0.8
f = open('output.txt', mode = 'w', encoding = 'utf-8')
f.write('|=====|\n')
f.write('::t:::dv(t):::v(t):::\n')
f.write('|=====|\n')
for t in range(11):
    f.write('| :3d | :7.4f | :8.4f |\n'.format(t,dv,v))
    t += 1
    v = v + dv
    dv = dv / 2
f.write('|=====|\n')
f.close()
print("Печать таблицы окончена...")
```

5. Первичный статистический анализ геологических данных

Цель первичного статистического анализа – построить статистическое распределение изучаемого параметра и вычислить числовые характеристики этого распределения.

Прежде всего, необходимо понять в какой шкале (номинальной, порядка, разностей, отношений) измерены данные. От этого зависит дальнейший анализ. Так, есть смысл строить статистическое распределение в стандартном понимании этого термина только для тех величин, которые измерены в шкалах разностей или отношений.

Допустим, что нам дана выборка значений, измеренных в **сильных** шкалах: $x_1, x_2, \dots, x_n$. Какие её характеристики нас интересуют?

Минимальное значение выборки: $x_{\min} = \min_{i=1 \dots n} x_i$.

Максимальное значение выборки: $x_{\max} = \max_{i=1 \dots n} x_i$.

Размах значений выборки: $w = x_{\max} - x_{\min}$.

Среднее выборочное: $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$. Данный параметр характеризует средний уровень данных.

Выборочная (несмещённая) дисперсия: $s^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$. Данный параметр характеризует среднее значение квадрата разброса данных около их среднего.

Выборочное (несмещённое) среднее квадратическое отклонение: $s = \sqrt{s^2}$. Данный параметр характеризует среднее значение разброса данных около их среднего.

Мода: это такое значение, которое появляется наиболее часто.

Если мы имеем данные, измеренные в сильных шкалах, и их много (не менее нескольких десятков, а лучше сотен) то для дальнейшего анализа их надо **группировать** и от исследования точечных значений переходить к исследованию интервалов. Промежуток от $x_{\min}$ до $x_{\max}$ необходимо разбить на части (обычно одинаковой длины) и выяснить относительную частоту попадания значений выборки в каждый частичный интервал. Таким образом, мы получим выборочное интервальное распределение.

Заметим, что один из способов вычисления количества интервалов – формула Стёрджесса: $k = 1 + \lfloor \log_2 n \rfloor$.

После вычисления количества интервалов можно определить длину интервалов: $\delta = \frac{w}{k}$.

Статистическое распределение можно представить в виде таблицы, в первом столбце которой помещаются интервалы разбиения:

$$[x_{\min}; x_{\min} + \delta], (x_{\min} + \delta; x_{\min} + 2\delta], \dots, (x_{\min} + (k-1)\delta; x_{\min} + k\delta],$$

причем $x_{\min} + k\delta = x_{\max}$, а во втором столбце помещаются относительные частоты p_i ($i = 1 \dots k$) попадания значений выборки в интервалы. Относительная частота равна отношению количества попадания в интервал к объёму выборки n . Заметим, что сумма всех относительных частот равна 1. Также заметим, что иногда левый конец первого интервала и правый конец последнего интервала чуть расширяют. Это делают для удобства подсчёта частот, чтобы концы интервалов не совпали ни с одним из выборочных значений.

Пример

Описание ситуации. Для определения петрографического типа пород из горизонта неогеновых лав отобрано и проанализировано на содержание SiO_2 $n = 30$ проб. Содержание SiO_2 в отдельных пробах меняется от 56.6 процентов (андезитобазальт) до 73.2 процентов (риолит), что не позволяет оценить состав лав в целом по единичному наблюдению.

Задание: загрузить выборочные данные из файла, вычислить характеристики выборки, построить таблицу статистического распределения, сохранить результаты в файл.

Данные:

№ пробы	$\text{SiO}_2(\%)$	№ пробы	$\text{SiO}_2(\%)$	№ пробы	$\text{SiO}_2(\%)$
1	59.5	11	62.9	21	71.4
2	66.8	12	62.4	22	67.6
3	60.5	13	71.6	23	63.6
4	63.7	14	65.8	24	61.1
5	72.5	15	63.1	25	63.8
6	69.2	16	61.2	26	67.5
7	61.2	17	69.3	27	65.3
8	66.3	18	64.6	28	69.9
9	73.2	19	67.8	29	73.2
10	64.6	20	56.6	30	60.7

Предварительно нужно создать текстовый файл `data.txt`, содержащий следующие данные (два столбца чисел, разделённых пробелом в каждой строке; первый столбец – это номера измерений, а второй – это собственно данные):

Листинг 1.

```
1 59.5
2 66.8
3 60.5
4 63.7
5 72.5
6 69.2
7 61.2
8 66.3
9 73.2
10 64.6
11 62.9
12 62.4
13 71.6
14 65.8
15 63.1
16 61.2
17 69.3
18 64.6
19 67.8
20 56.6
21 71.4
22 67.6
23 63.6
24 61.1
25 63.8
26 67.5
```

27 65.3
28 69.9
29 73.2
30 60.7

Программа по обработке этих данных может иметь вид (в программе использован модуль statistics, если он не установлен, то нужно его установить):

Листинг 2.

```
f = open('data.txt', 'r')
dataFrame = f.readlines()
f.close()
f = open('output.txt', 'w')
def ListToIntAndFloat(lst):
    return [int(lst[0]),float(lst[1])]
dataFrame = [ListToIntAndFloat(item.strip().split(sep=' ')) for item in dataFrame]
data = [item[1] for item in dataFrame]
n = len(dataFrame)
f.write('Объём выборки: '+str(n)+'\n')
xMax = max(data)
f.write('Максимальное значение: '+str(xMax)+'\n')
xMin = min(data)
f.write('Минимальное значение: '+str(xMin)+'\n')
w = xMax - xMin
f.write('Размах: '+str(w)+'\n')
xBar = sum(data)/n
f.write('Среднее выборочное: {:.3f}\n'.format(xBar))
sSquared = sum([(x-xBar)**2 for x in data])/(n-1)
f.write('Выборочная дисперсия: {:.3f}\n'.format(sSquared))
import math
s = math.sqrt(sSquared)
f.write('Среднеквадратичное отклонение: {:.3f}\n'.format(s))
import statistics
try:
    xMod = statistics.mode(data)
    f.write('Мода: '+str(xMod)+'\n')
except Exception:
    f.write('Какой-то косяк с модой...\n')
k = 1+math.floor(math.log(n,2))
f.write('Число интервалов: '+str(k)+'\n')
xMin = xMin - 0.0001
xMax = xMax + 0.0001
w = xMax - xMin
d = w/k
f.write('Длина частичного интервала: {:.3f}\n'.format(d))
p = []
for i in range(k):
    j = 0
    for item in data:
        if xMin+i*d < item <= xMin+(i+1)*d:
            j += 1
    p.append(j)
```

```

f.write('|=====|\n')
f.write('| интервал |частота|\n')
f.write('|=====|\n')
for i in range(k):
    f.write('|({:0.3f};{:0.3f}] |{:0.3f} |\n'.format(xMin+i*d,xMin+(i+1)*d,p[i]/n))
f.write('|=====|\n')
f.close()
print('Программа закончила работу...')

```

При выполнении программа создаст файл с результатами исследования:

Листинг 3.

```

Объём выборки: 30
Максимальное значение: 73.2
Минимальное значение: 56.6
Размах: 16.6
Среднее выборочное: 65.563
Выборочная дисперсия: 19.180
Среднеквадратичное отклонение: 4.379
Какой-то косяк с модой...
Число интервалов: 5
Длина частичного интервала: 3.320
|=====|
| интервал |частота|
|=====|
|(56.600;59.920]| 0.067 |
|(59.920;63.240]| 0.267 |
|(63.240;66.560]| 0.267 |
|(66.560;69.880]| 0.200 |
|(69.880;73.200]| 0.200 |
|=====|

```
