Информатика. Кластеризация. Функции.

Содержание

1 Кластеризация

В данном уроке всё время будет уделено именно кластеризации – разделению звезд по кластерам. Нахождение центроидов и среднего арифметического ничем не отличается от того, что уже было пройдено, поэтому для самопроверки это можно сделать самостоятельно.

1.1 Пример 1

Учёный решил провести кластеризацию некоторого множества звёзд по их расположению на карте звёздного неба. Кластер звёзд – это набор звёзд (точек) на графике, лежащий внутри круга радиусом R. Каждая звезда обязательно принадлежит только одному из кластеров.

Истинный центр кластера, или центроид, – это одна из звёзд на графике, сумма расстояний от которой до всех остальных звёзд кластера минимальна. Центроид не вычисляется для колец, он вычисляется только для кластеров, представляющих собой круг.

Под расстоянием понимается расстояние Евклида между двумя точками $A(x_1, y_1)$ и $B(x_2, y_2)$ на плоскости, которое вычисляется по формуле:

$$d(A, B) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

В файле А хранятся данные о звёздах **двух** кластеров, где $R = 1,5$ для внутреннего кластера и $R = 3,1$ для внешнего кластера. В каждой строке записана информация о расположении на карте одной звезды: сначала координата x, затем координата y. Значения даны в условных единицах, которые представлены вещественными числами. Известно, что количество звёзд не превышает 1245.

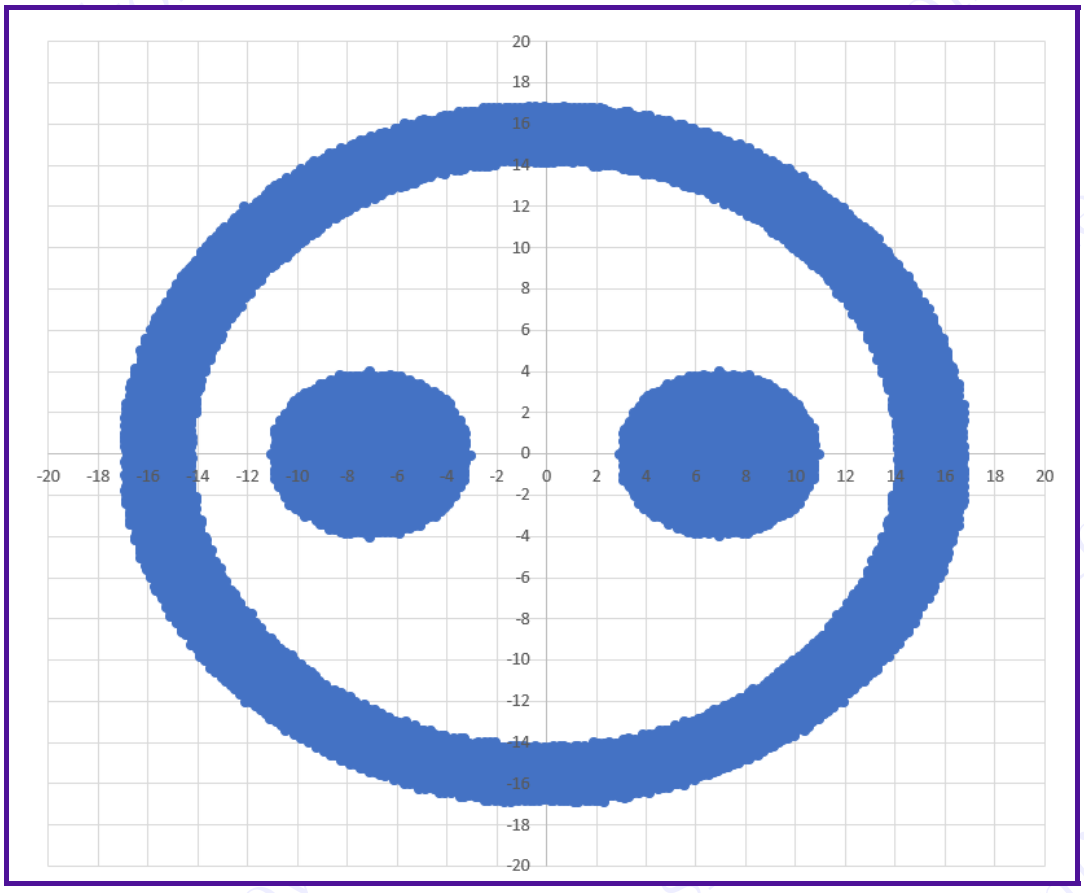
В файле Б хранятся данные о звёздах **трёх** кластеров, где $R = 4,1$ для двух внутренних кластеров и $R = 17,1$ для внешнего кластера. Известно, что количество звёзд не превышает 9414. Структура хранения информации о звездах в файле Б аналогична файлу А.

Для каждого файла определите координаты центра каждого кластера, затем вычислите два числа: P_x — среднее арифметическое абсцисс центров кластеров, и P_y – среднее арифметическое ординат центров кластеров.

В ответе запишите четыре числа через пробел: сначала целую часть произведения $P_x \cdot 100$ для файла А и $P_y \cdot 100$ для файла А, далее целую часть произведения $P_x \cdot 100$ для файла Б и $P_y \cdot 100$ для файла Б.

Решение

Для начала обязательно открываем файл в *Excel* и строим заданные нам кластеры:



Из рисунка видно, что нас истересуют две окружности, которые лежат в прямоугольнике, в котором $-12 < x < 12, -5 < y < 5$. Причем первая окружность находится в отрицательных абсциссах, а вторая – в положительных.

Реализуем программу, разделяющую звезды на два кластера по заданным условиям:

```
f = open('27-1.txt') # Открываем файл для чтения
s = f.readline() # Читаем первую строку файла (X и Y)
# Создаем список a, где каждый элемент - список из двух чисел
# Заменяем запятые на точки и разбиваем строку на числа
a = [list(map(float, i.replace(',','.').split())) for i in f]
clusters = [[], []] # Создаем два пустых кластера
for i in a: # Разбиваем точки на два кластера по условиям
    x, y = i
    # Проверяем, попадает ли точка в заданный прямоугольник
    if (y < 5) and (y > -5) and (x < 12) and (x > -12):
        if x < 0: # Если x отрицательный, определяем в первый кластер
            clusters[0].append(i)
        else: # Иначе - во второй кластер
            clusters[1].append(i)
```

С помощью модуля Черепаха проверим, что точки распределились правильно:

```
from turtle import * # Импортируем библиотеку turtle для визуализации
tracer(0), m = 20 # Отключаем анимацию и определяем масштаб
pu() # Поднимаем перо
for x in range(-200, 200): # Рисуем фон - фиолетовые точки в заданной области
    for y in range(-200, 200): # Проверяем те же условия, что и для кластеров
        if (y/5 < 5) and (y/5 > -5) and (x/5 < 12) and (x/5 > -12):
            goto(x / 5 * m, y / 5 * m)
            dot(5, 'purple')  # Рисуем фиолетовую точку размером 5
for i in clusters[0]: # Рисуем точки из первого кластера
    x, y = i
    goto(x*m, y*m) # Перемещаемся к точке с учетом масштаба
    dot(5) # Рисуем точку размером 5
```

1.2 Пример 2

Учёный решил провести кластеризацию некоторого множества звёзд по их расположению на карте звёздного неба. Кластер звёзд – это набор звёзд (точек) на графике, каждая из которых находится от хотя бы одной другой звезды на расстоянии не более R условных единиц. Каждая звезда обязательно принадлежит только одному из кластеров. Истинный центр кластера, или центроид, – это одна из звёзд на графике, сумма расстояний от которой до всех остальных звёзд кластера минимальна. Под расстоянием понимается расстояние Евклида между двумя точками $A(x_1, y_1)$ и $B(x_2, y_2)$ на плоскости, которое вычисляется по формуле:

$$d(A, B) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

В файле А хранятся данные о звёздах двух кластеров, где $R = 0,2$ для каждого кластера. В каждой строке записана информация о расположении на карте одной звезды: сначала координата x, затем координата y. Значения даны в условных единицах, которые представлены вещественными числами. Известно, что количество звёзд не превышает 2000.

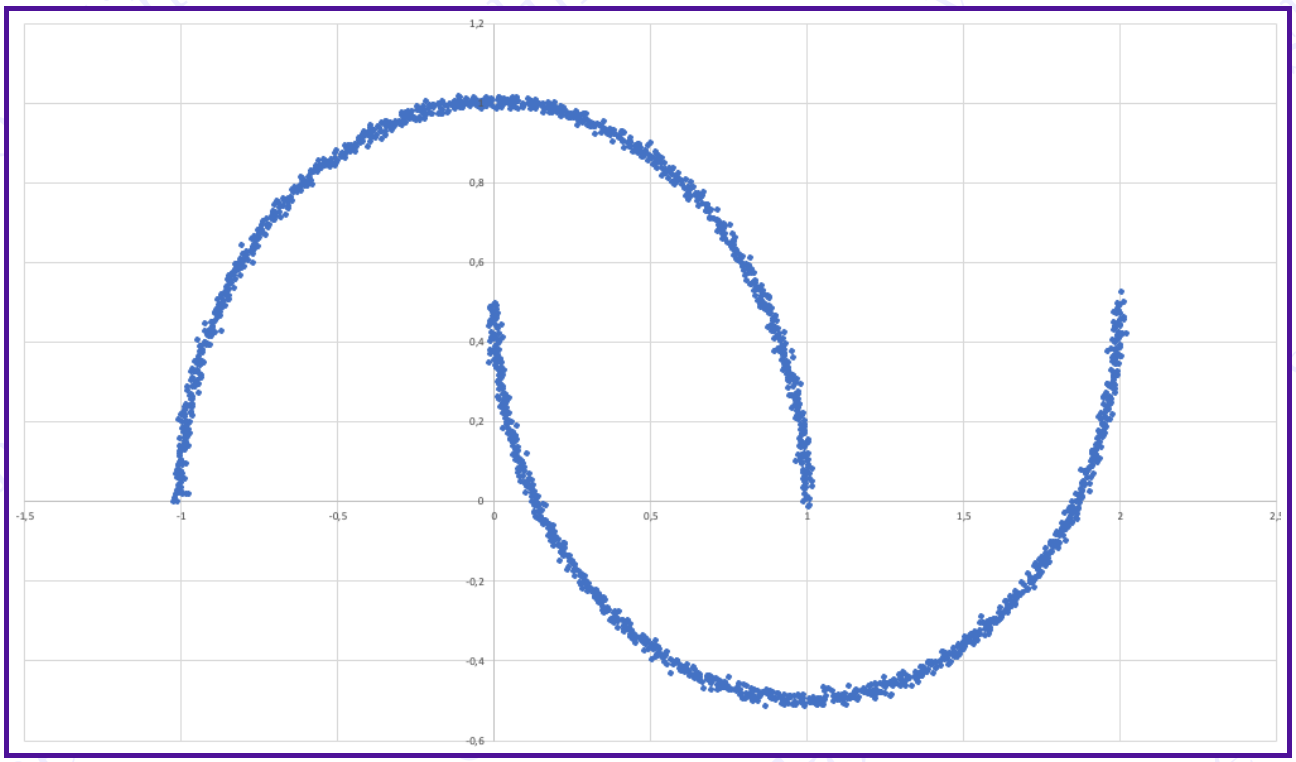
В файле Б хранятся данные о звёздах четырех кластеров, где $R = 0,2$ для каждого кластера. Известно, что количество звёзд не превышает 10 000. Структура хранения информации о звездах в файле Б аналогична файлу А.

Для каждого файла определите координаты центра каждого кластера, затем вычислите два числа: P_x — среднее арифметическое абсцисс центров кластеров, и P_y – среднее арифметическое ординат центров кластеров.

В ответе запишите четыре числа через пробел: сначала целую часть произведения $P_x \cdot 10000$ для файла А и $P_y \cdot 10000$ для файла А, далее целую часть произведения $P_x \cdot 10000$ для файла Б и $P_y \cdot 10000$ для файла Б.

Решение

Для начала обязательно открываем файл в *Excel* и строим заданные нам кластеры:



Отделять точки будем с помощью неравенств окружностей. Для этого вспомним формулу, по которой строится окружность:

$$(x - x_0)^2 + (y - y_0)^2 = R^2$$

Левая окружность имеет примерный центр $(0, 0)$ и радиус 1.3. Правая окружность – центр $(1, 0.5)$ и радиус 1.3.

Также учтем, что ординаты левой окружности превышают -0.2, а правой – не превышают 0.6.

Реализуем программу, разделяющую звезды на два кластера по заданным условиям:

```
f = open('27-2.txt') # Открываем файл для чтения
s = f.readline() # Читаем первую строку файла (X и Y)
# Создаем список a, где каждый элемент - список из двух чисел
# Заменяем запятые на точки и разбиваем строку на числа
a = [list(map(float, i.replace(',','.').split())) for i in f]
clusters = [[], []] # Создаем два пустых кластера
for i in a: # Распределяем точки по кластерам согласно условиям
    x, y = i
    if (x*x + y*y <= 1.3) and (x*x + y*y >= 0.6) and (y > -0.2):
        clusters[0].append(i)
    elif ((x-1)**2 + (y-0.5)**2 <= 1.3) and ((x-1)**2 + (y-0.5)**2 >= 0.6)
    and (y < 0.6):
        clusters[1].append(i)
```

С помощью модуля Черепаха проверим, что точки распределились правильно:

```
from turtle import * # Импортируем библиотеку turtle для визуализации
tracer(0) # Отключаем анимацию
pu() # Поднимаем перо
m = 200 # Масштаб
for i in clusters[0]: # Рисуем точки из первого кластера
    x, y = i
    goto(x*m, y*m) # Перемещаемся к точке с учетом масштаба
    dot(5) # Рисуем точку размером 5 пикселей
```