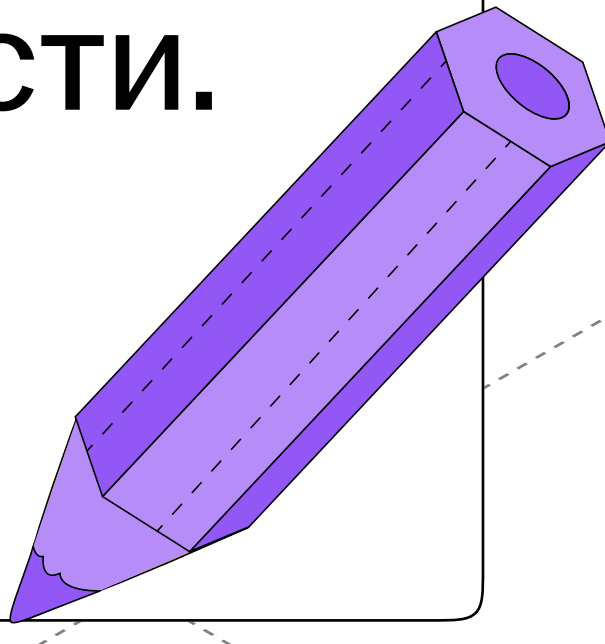


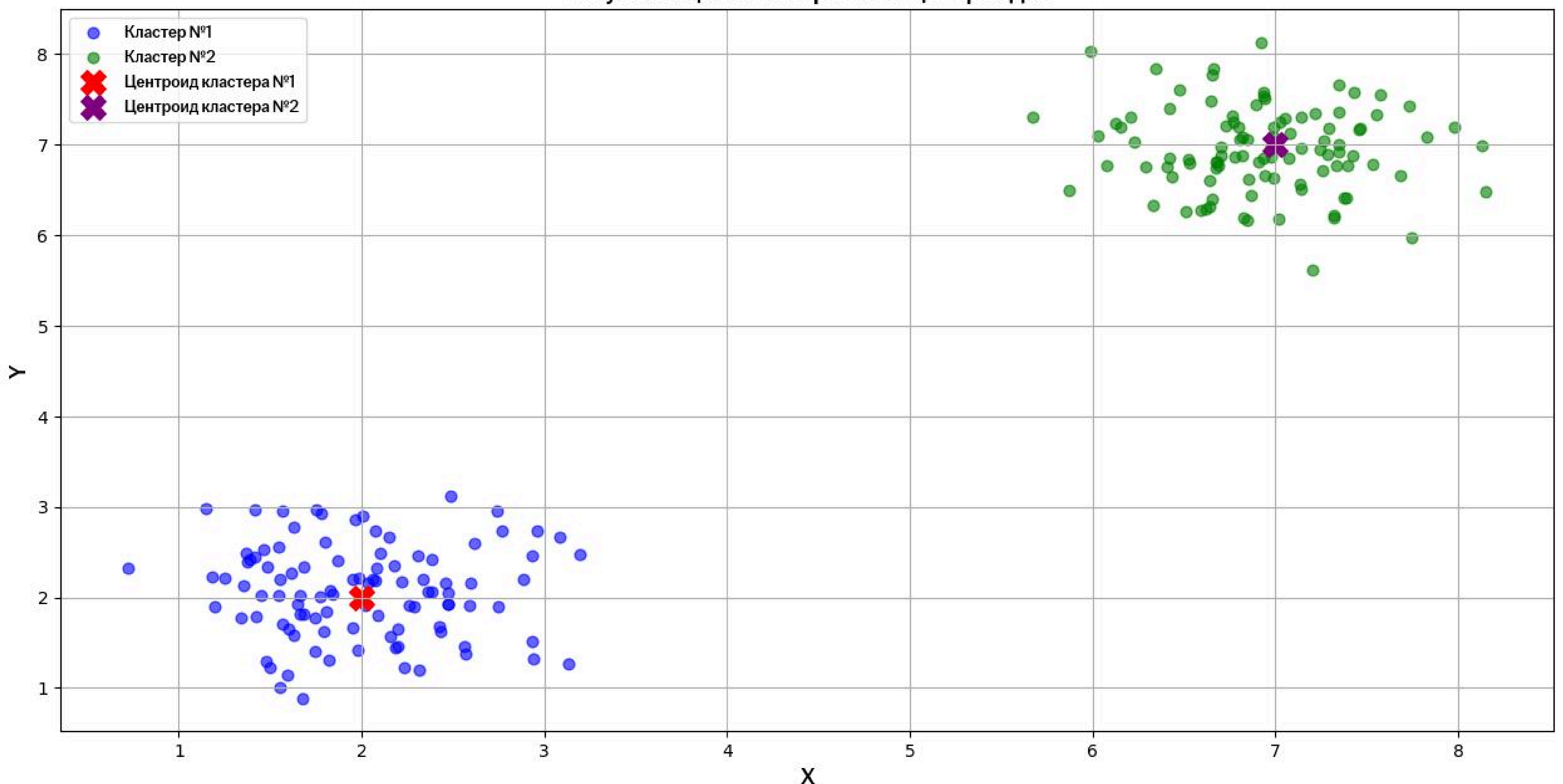
Визуализация алгоритма кластеризации

Кластер – это группа объектов (например, звёзд, документов, клиентов и т.д.), которые находятся в пределах определённой области и схожи по каким-то признакам. Кластер может быть представлен в виде набора точек на плоскости или в многомерном пространстве, которые сгруппированы на основе расстояний или других мер схожести. В контексте задачи о звёздах кластером считается область, содержащая звёзды, расположенные близко друг к другу в рамках заданного прямоугольника.



Центроид кластера – это центральная точка, которая характеризует данный кластер. В случае задачи о звёздах центроид выбирается среди самих звёзд, и для его нахождения вычисляется точка, от которой сумма расстояний до всех остальных звёзд кластера минимальна.

Визуализация кластеров и их центроидов



Кластеризация – это метод машинного обучения, используемый для группировки данных в подмножества (кластеры), основываясь на схожести объектов внутри каждого кластера.

Основная цель кластеризации – разделить набор данных так, чтобы объекты внутри одного кластера были схожи, а между кластерами – максимально различались. В отличие от классификации, кластеризация относится к методам обучения без учителя, так как заранее не предполагается, что данные уже размечены (разделены на классы).



Визуализация алгоритма кластеризации

Функции и методы модуля turtle

`penup` — включает режим бесследного перемещения исполнителя.

`pen` — включает режим, позволяющий оставлять Черепахе след в виде линии.

`forward (n)` — вызывает передвижение Черепахи на `n` шагов в направлении движения.

`backward (k)` — вызывает передвижение Черепахи на `k` шагов против направления движения.

`left (a)` — изменяет направление движения на `a` градусов против часовой стрелки.

`right (b)` — изменяет направление движения на `b` градусов по часовой стрелке.

`dot(size, color)` — рисует круг заданного размера и цвета.

Функция визуализации итогов кластеризации

```
from turtle import *

def viz_clusters(clusters):
    screensize(2000, 2000)
    penup(), tracer(0)
    colors = ['red', 'blue']
    for i in range (len(clusters)):
        for x, y in clusters[i]:
            goto(x * 100, y * 100)
            dot(5, colors[i])
    done()
```

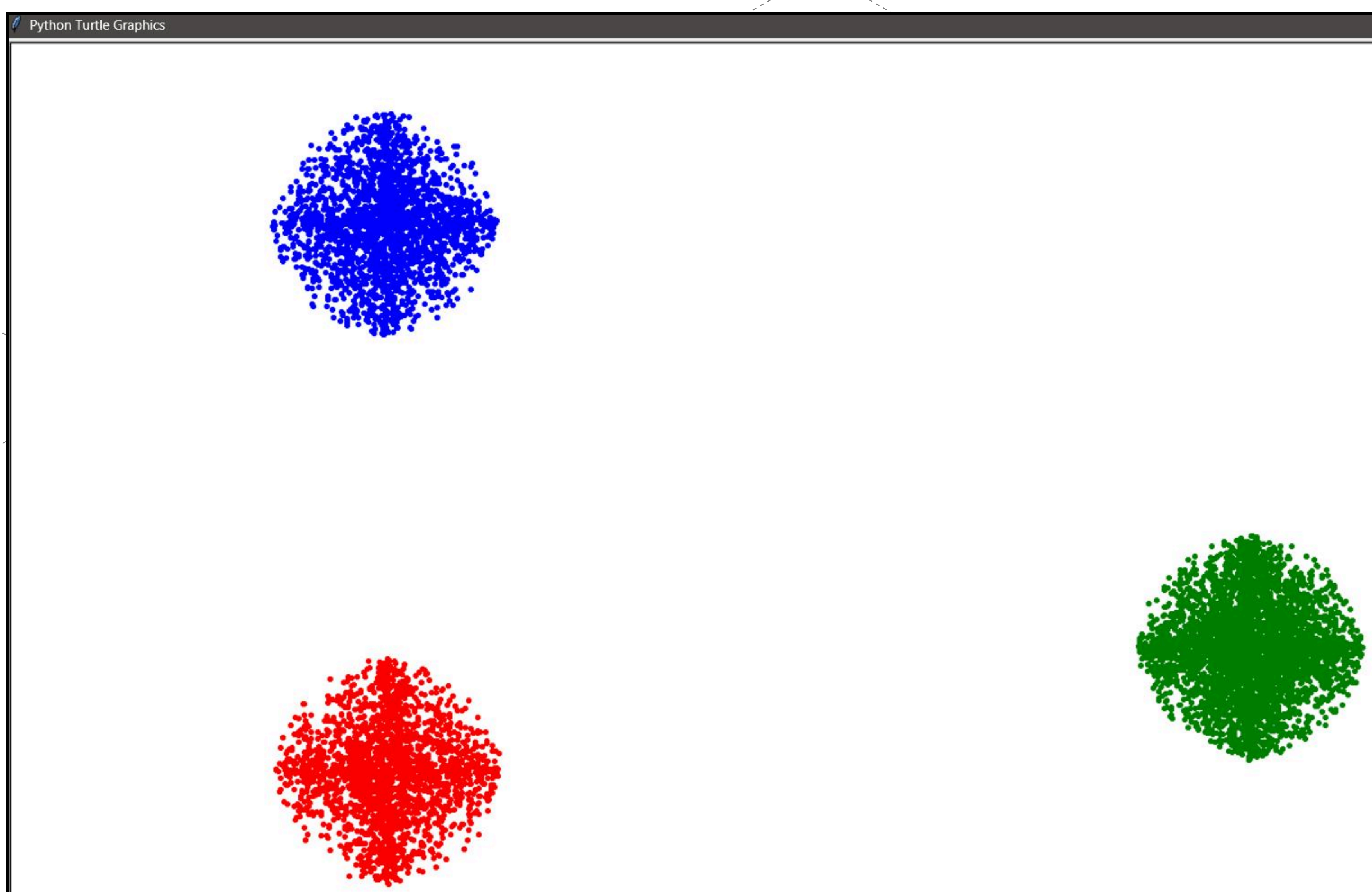
1. Цветовая дифференциация кластеров:

- Для каждого кластера используется свой цвет.
- Цвет задается через массив `colors`, индекс которого соответствует номеру кластера.

2. Рисование точек:

- `dot(5, colors[i])` рисует точку с координатами `(x, y)` в текущем кластере.
- Позиция точки предварительно вычисляется и масштабируется, чтобы точки лучше размещались на графике.

Общая идея: мы проходим по всем точкам каждого кластера и рисуем их в нужном месте и цвете, чтобы визуально разделить данные.



По итогу получаем некую визуализацию того, как наш алгоритм распределил точки по кластерам.

В случае, если в процесса кластеризации мы ошиблись, мы сможем увидеть.

Как это увидеть? Точки разных цветов будут находиться в одном скоплении точек (кластере), а значит в данном случае была допущена ошибка.

На примере ошибок нет, а значит точки правильно распределены по кластерам.



Заметки

