实验十：聚类算法

姓名：　　学号：

- 实验目的

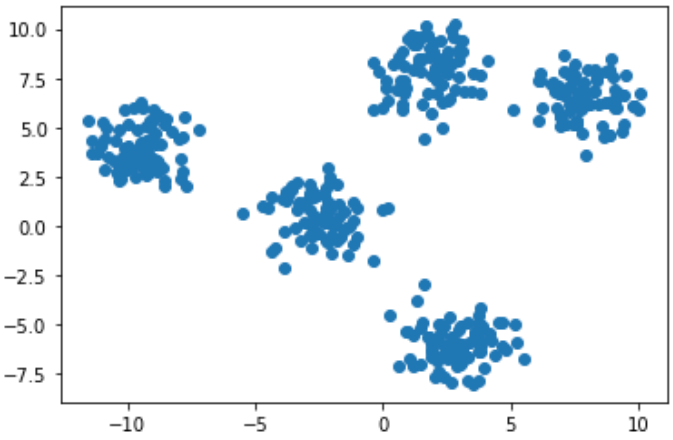
了解无监督任务范式概念，掌握聚类思想，掌握 k-means 算法基本原理和实现方法。

- 实验要求

编程实现 k 均值聚类算法，对如下数据进行聚类。对于 k 均值算法，随机从样本中选出 k 个点作为初始聚类中心，并设置迭代次数为 100。依次将聚类数设置为$k = 1,2,3,\ldots,10$，计算相应聚类结果的簇内平方误差指标。

$$\text{loss} = \sum_{i=1}^{k} \sum_{\boldsymbol{x} \in C_i} \left\| \boldsymbol{x} - \boldsymbol{u}^{(i)} \right\|^2$$

绘制不同k值时聚类结果图，用不同颜色表示不同的类。绘制 loss 值随k值增加的变化曲线图。



- 实验环境

Python, numpy, matplotlib

- 实验代码

```
import numpy as np

import random

from matplotlib import pyplot as plt
```

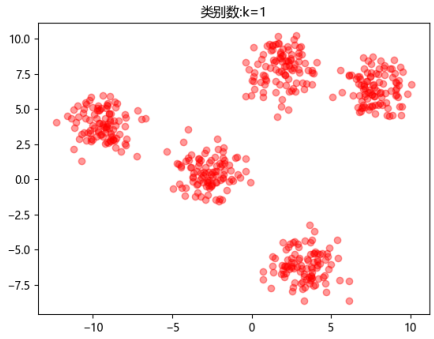
```
data = np.loadtxt('experiment_10_training_set.csv', delimiter=',')


loss = []
for m in range(10):
    center_index = random.sample(range(data.shape[0]), m + 1)
    center = data[center_index, 0:2]
    tag = np.zeros([data.shape[0], 1])
    data_tag = np.hstack((data, tag))
    for k in range(100):
        for i in range(data_tag.shape[0]):
            min_distance = np.inf
            min_tag = None
            for j in range(center.shape[0]):
                distance = np.square(data[i][0] - center[j][0]) + np.square(data[i][1] -
center[j][1])
                if distance < min_distance:
                    min_distance = distance
                    min_tag = j
            data_tag[i][2] = min_tag
        for j in range(center.shape[0]):
            tag_array = data_tag[np.where(data_tag[:, -1] == j)[0]]
            center[j] = np.mean(tag_array[:, 0:2], axis=0)
    colors = ['r', 'g', 'b', 'y', 'c', 'm', 'teal', 'darkorange', 'purple', 'black']
```

```
    plt.rcParams['font.sans-serif'] = ['Microsoft YaHei']
    for j in range(center.shape[0]):
        tag_array = data_tag[np.where(data_tag[:, -1] == j)[0]]
        plt.scatter(tag_array[:, 0], tag_array[:, 1], c=colors[j], alpha=0.4)
        plt.title(f"类别数:k={m + 1}")
    plt.show()
    loss_array = np.zeros([data_tag.shape[0], 1])
    center_indices = data_tag[:, 2].astype(int)
    distances_squared = ((data_tag[:, :2] - center[center_indices, :]) ** 2).sum(axis=1)
    loss_number = np.sum(distances_squared)
    loss = np.append(loss, loss_number)
print(loss)
plt.plot(range(1, 11), loss, marker='o')
plt.xlabel("k")
plt.ylabel("loss")
plt.title("loss 随 k 值增加的变化曲线图")
plt.show()
```

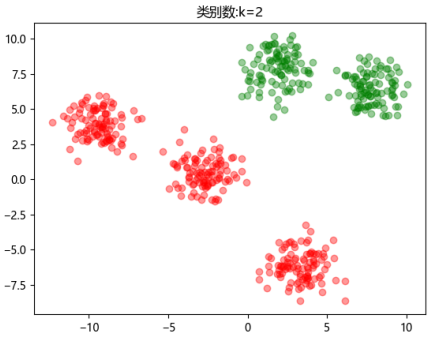
- 结果分析

(1) 聚类结果图

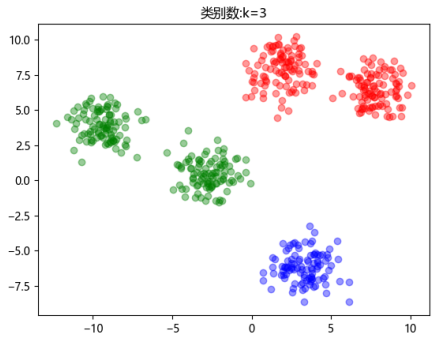
$k = 1$



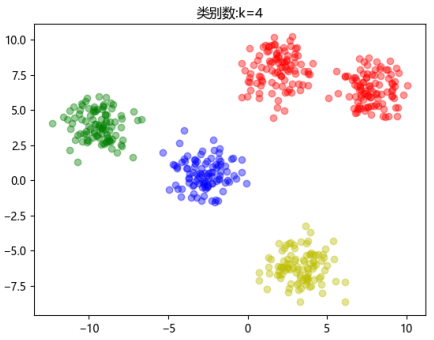
$k = 2$



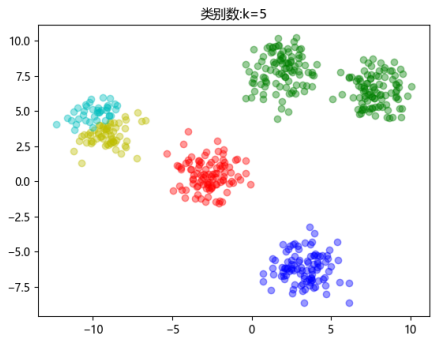
$k = 3$



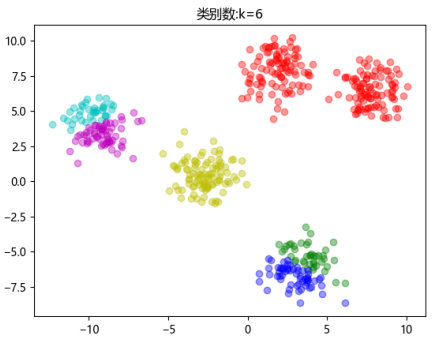
$k = 4$



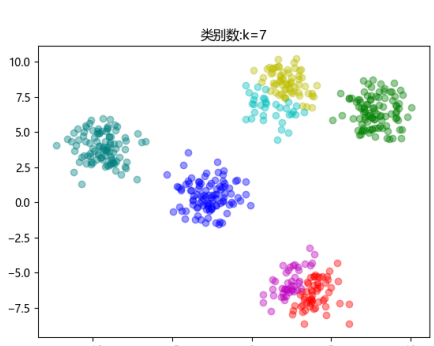
$k = 5$



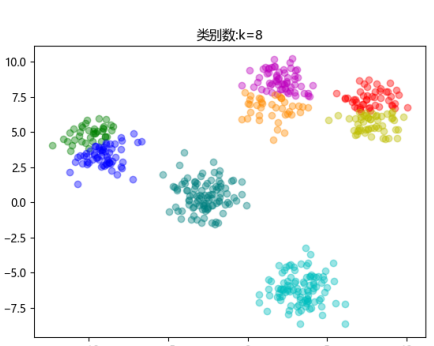
$k = 6$

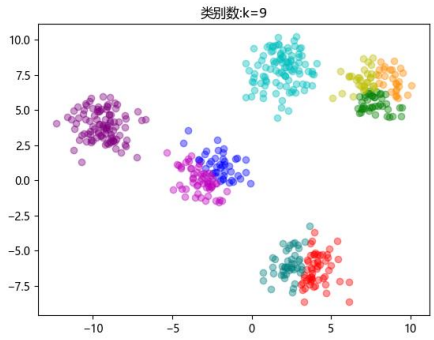
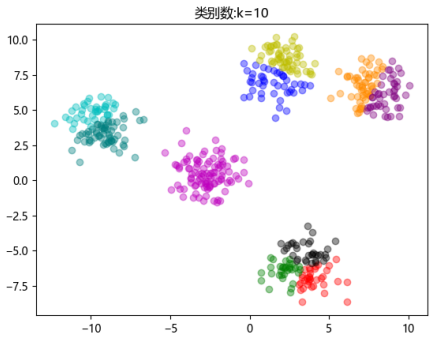


$k = 7$

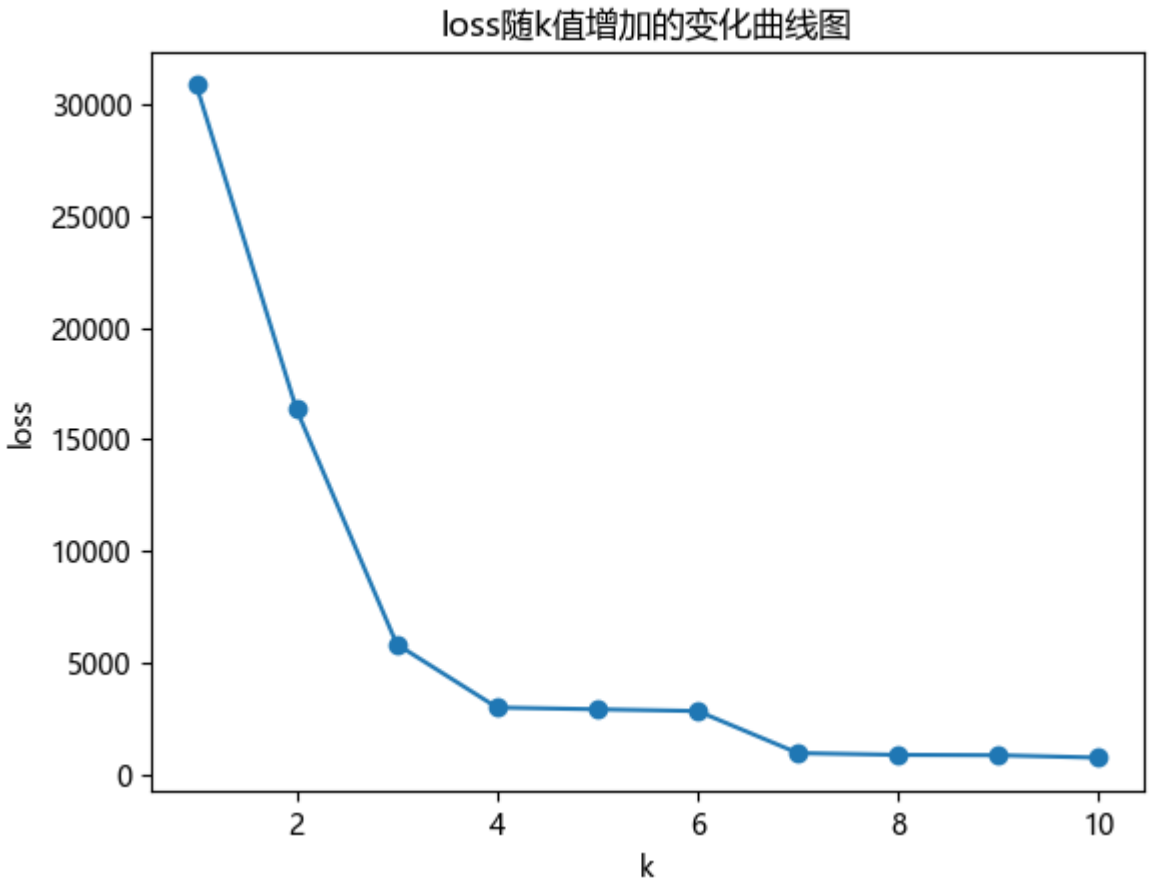


$k = 8$



$k=9$	$k=10$
	

(2) loss 随k值增加的变化曲线图



[30877. 842832　　16343. 73987033　5827. 892622　　2994. 959089

2914. 86833441　2837. 16846546　　953. 19066446　　874. 93215286

858. 40635967　　749. 68772305]