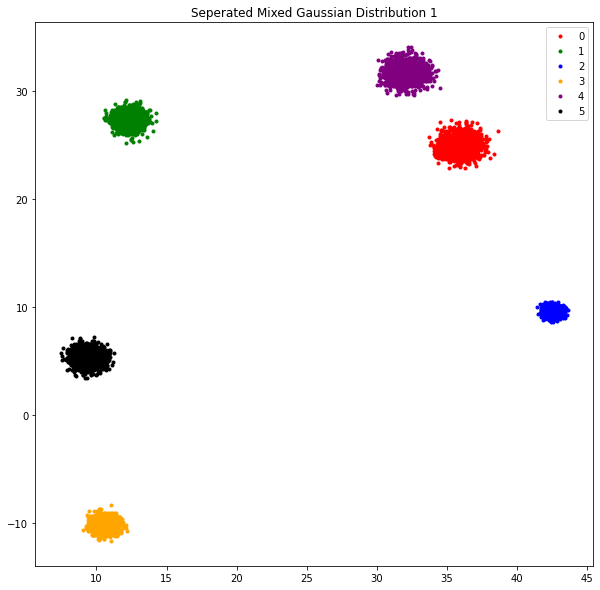
**Report**

自硕21 崔晏菲 2021210976

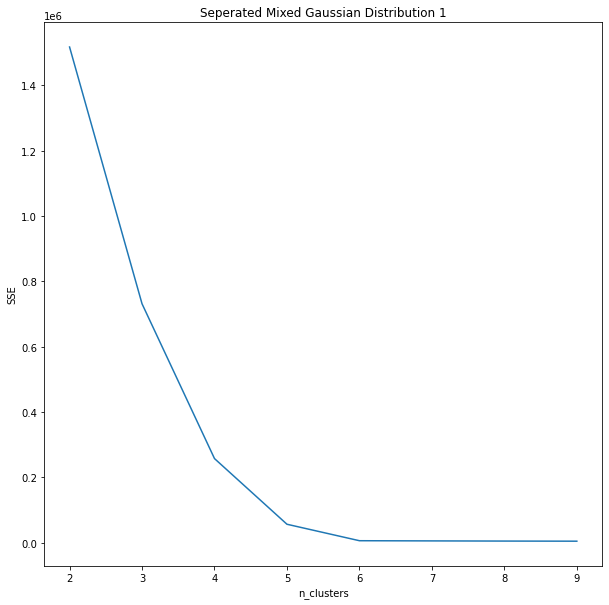
**1. 分离的混合高斯分布数据**

**Data 1:**

1. 数据可视化

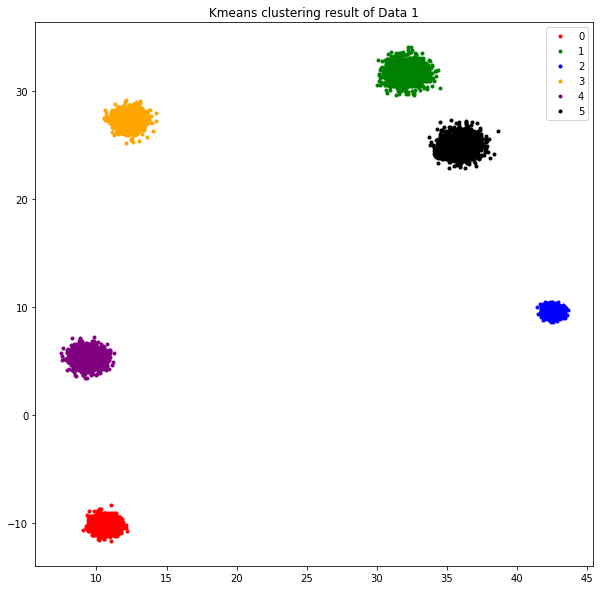


2. 碎石图



可见，当k = 6时，SSE减小为0，故6就是最佳聚类数。

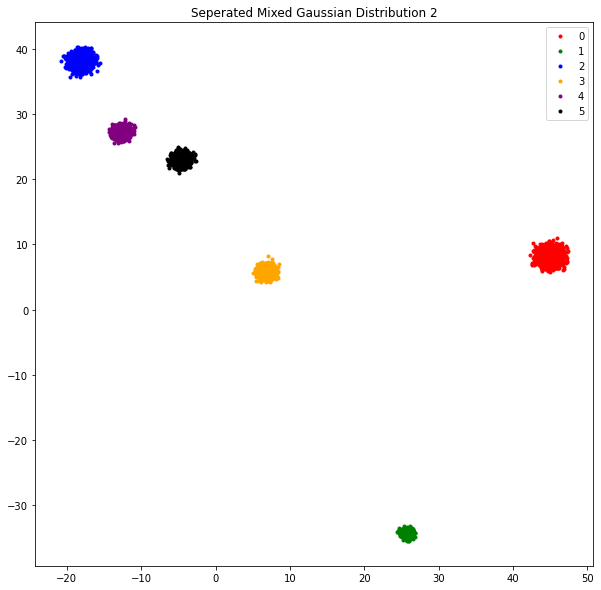
3. K-Means聚类结果



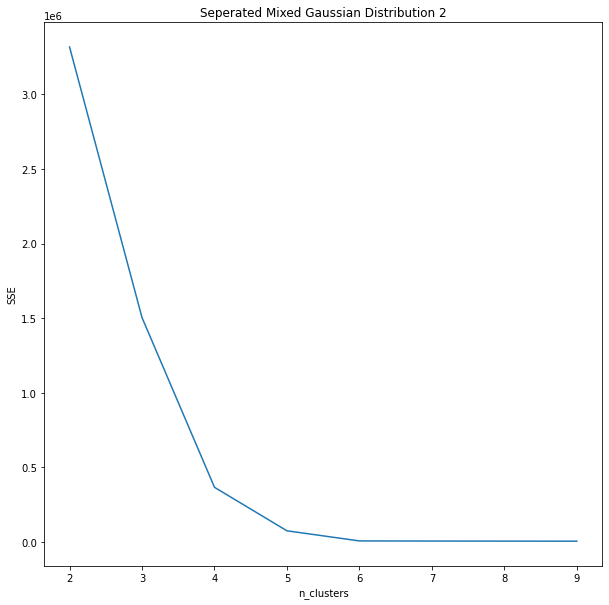
可见，聚类结果非常好，将所有的点都正确聚类了。

**Data 2:**

1. 数据可视化

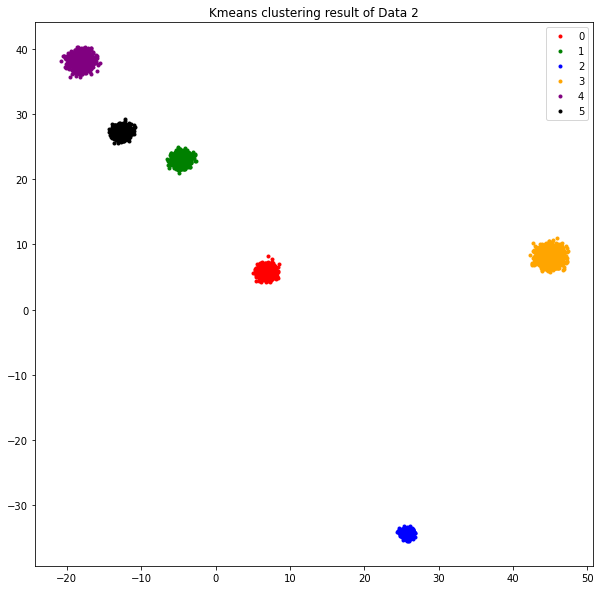


2. 碎石图



可见，当k = 6时，SSE减小为0，故6就是最佳聚类数。

3. K-Means聚类结果

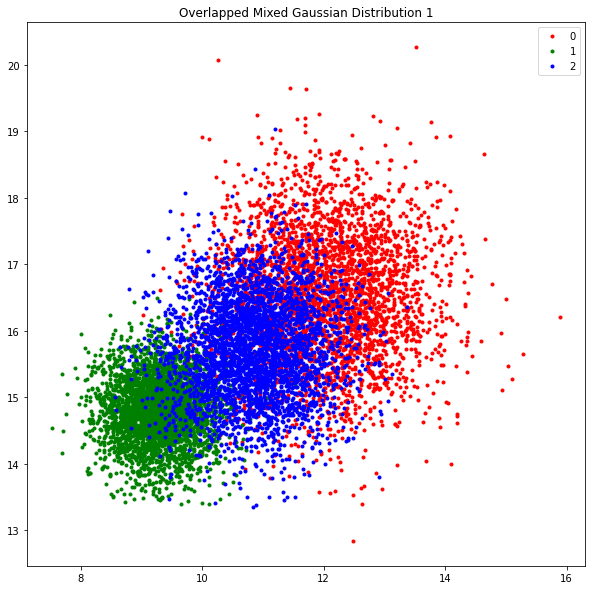


可见，聚类结果非常好，将所有的点都正确聚类了。

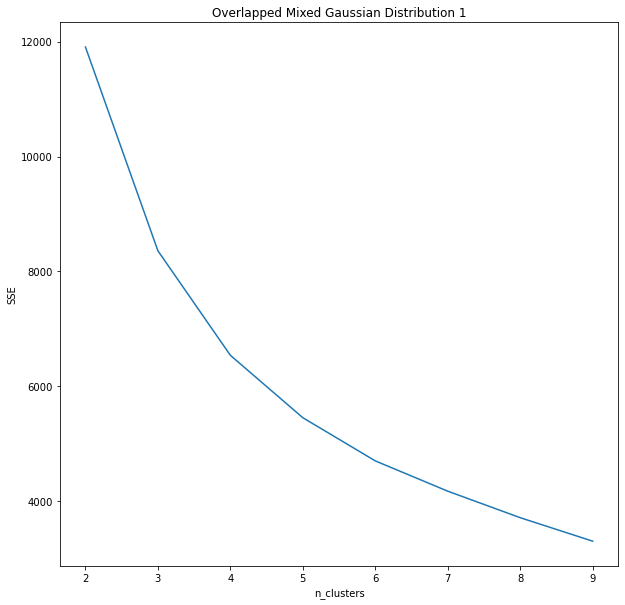
**2. 重叠的混合高斯分布数据**

**Data 3:**

1. 数据可视化

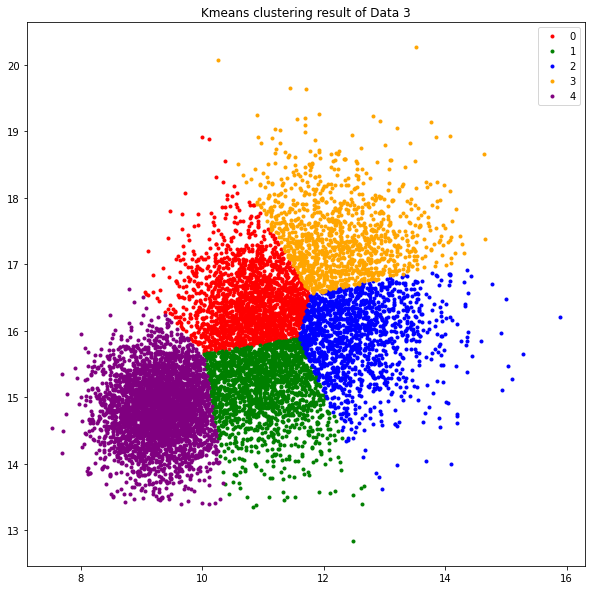


2. 碎石图



可见，由于数据重叠程度很大，并没有明显的拐点。我在这里选择拐点为k = 5。

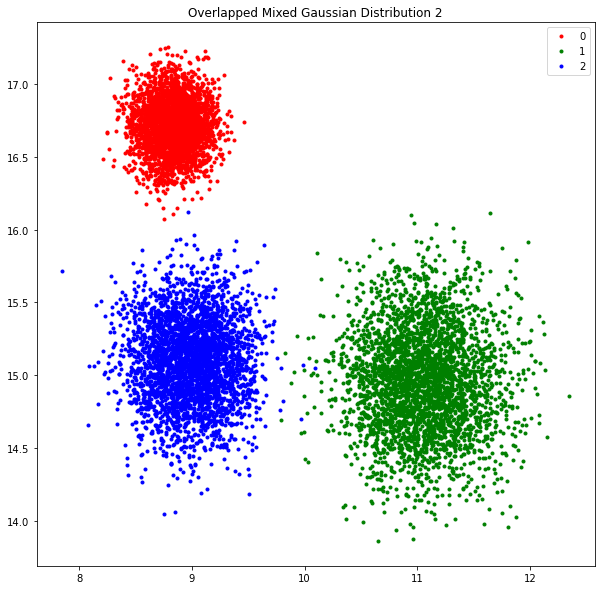
3. K-Means聚类结果



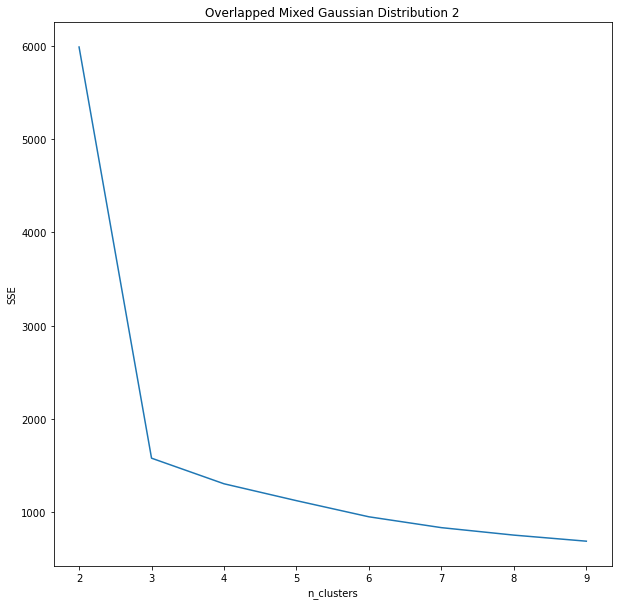
可见，我们可以非常看到，kmeans是典型的“硬”聚类，强行以欧氏距离计算。在这里，只有紫色部分的聚类结果稍微接近真实，其他的都无法辨别出来。

**Data 4:**

1. 数据可视化

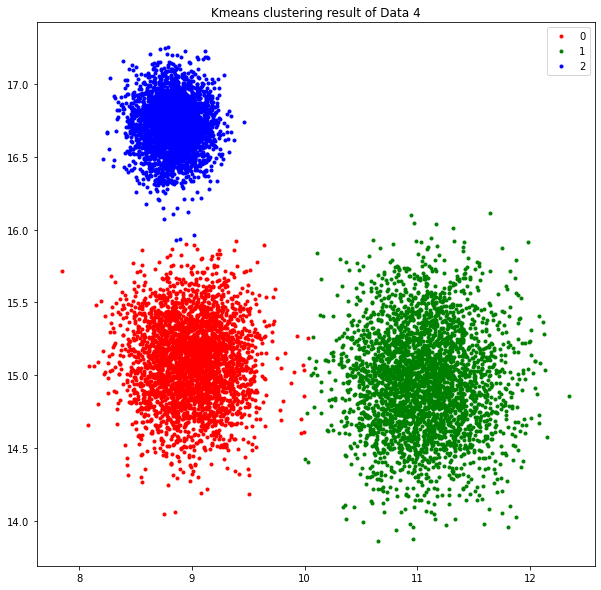


2. 碎石图



可见，因为数据类间距离稍微增大，k-means方法就可以非常好地获得拐点。

3. K-Means聚类结果

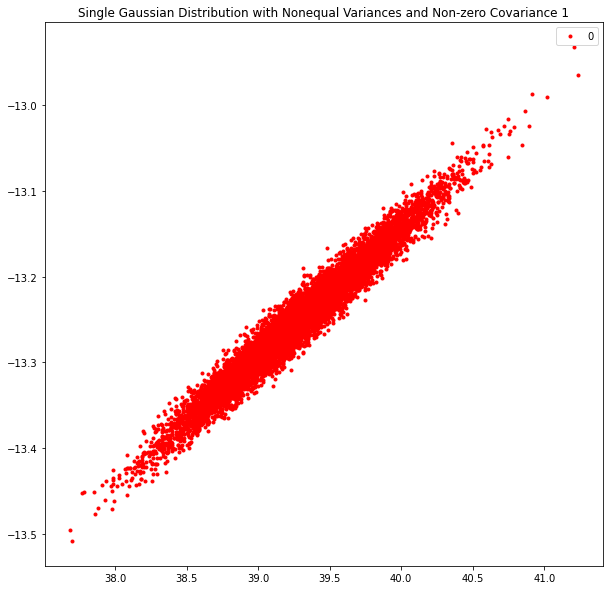


聚类结果整体上是准确的，只有每类边缘处的点稍微有错误。

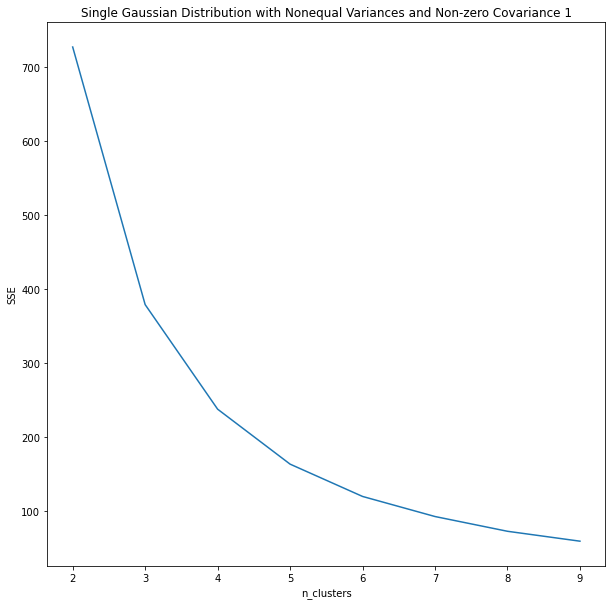
**3. 维度方差不同且协方差非零的单一高斯分布数据**

**Data 5:**

1. 数据可视化

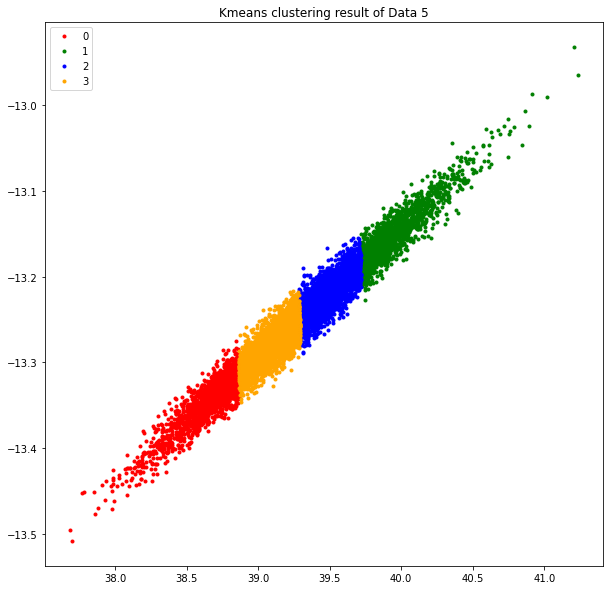


2. 碎石图



可见，因为是单个高斯分布数据，kmeans方法很难找到准确的聚类数。在这里我取k=4。

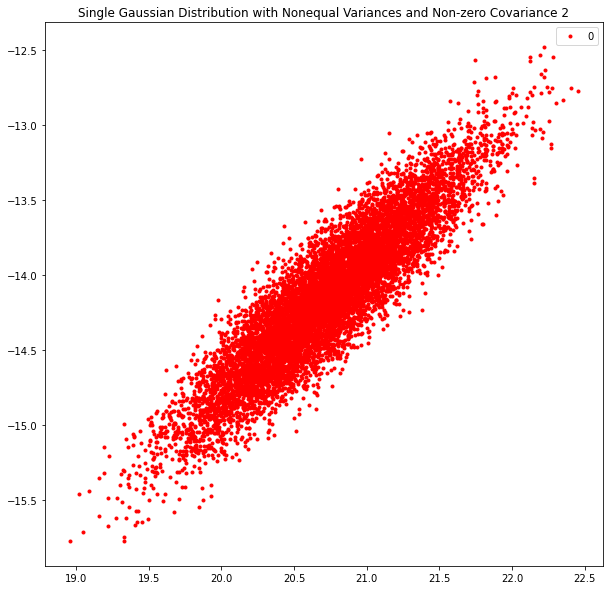
3. K-Means聚类结果



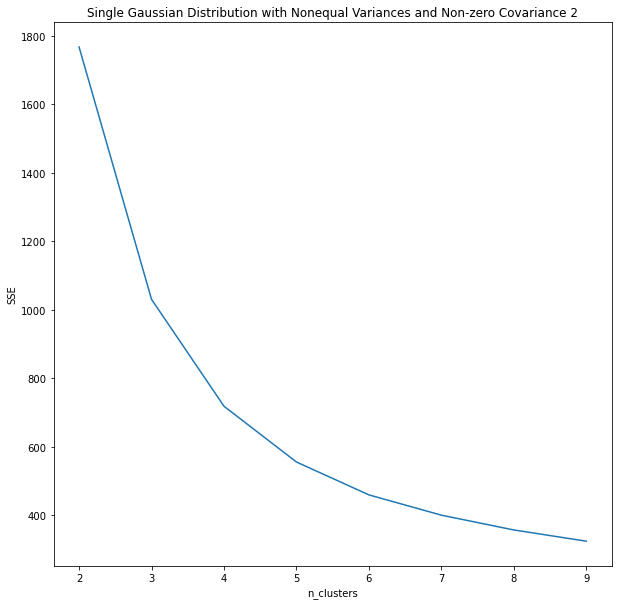
可见，kmeans在横轴方向上进行了聚类，因为横轴方向的方差更大。

**Data 6:**

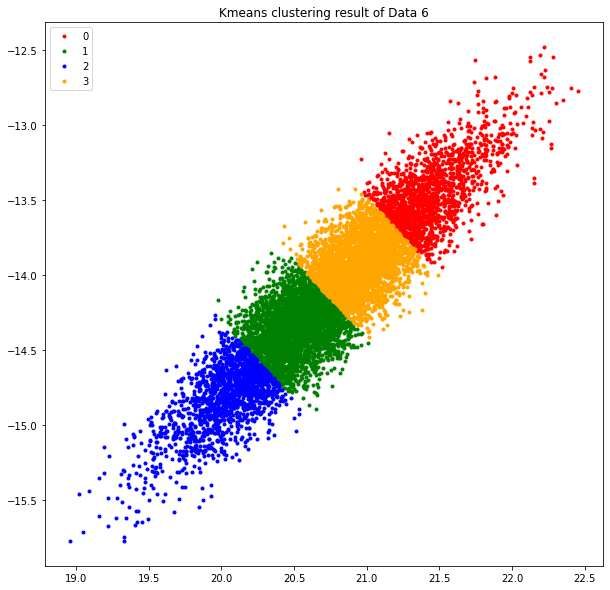
1. 数据可视化



2. 碎石图



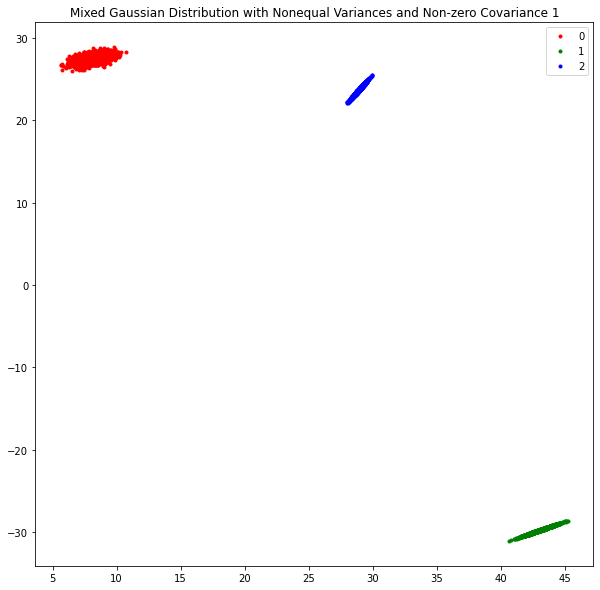
3. K-Means聚类结果



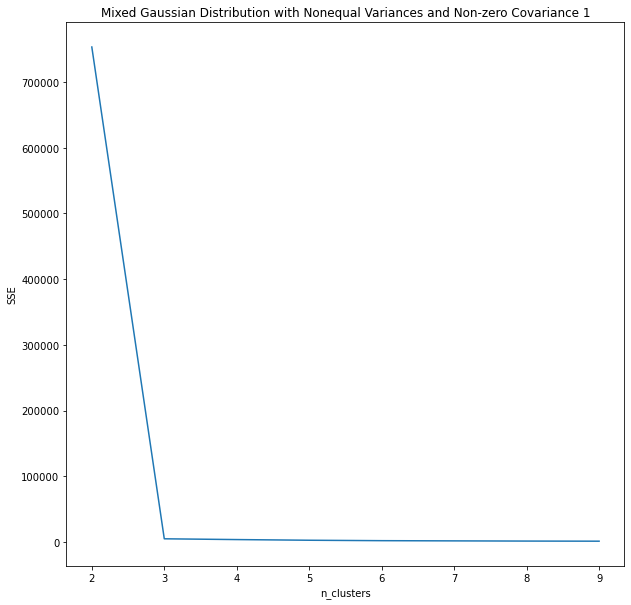
**4. 维度方差不同且协方差非零的混合高斯分布数据**

**Data 7:**

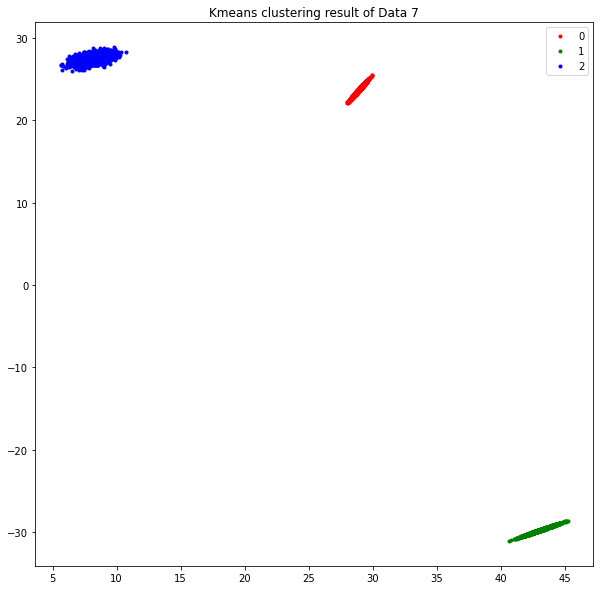
1. 数据可视化



2. 碎石图

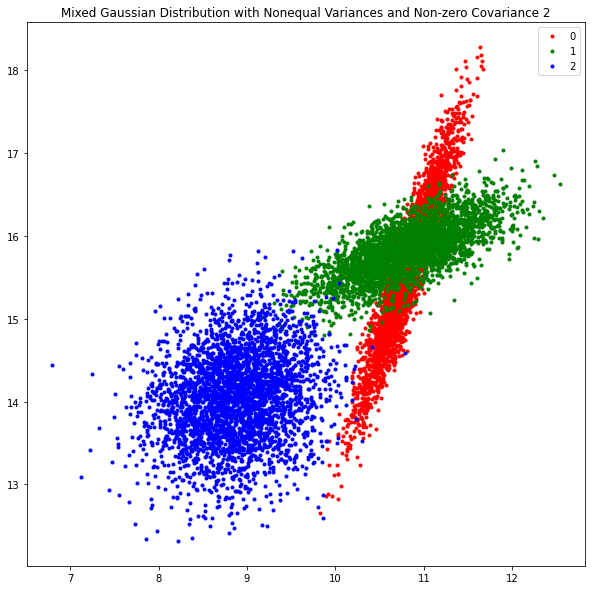


3. K-Means聚类结果



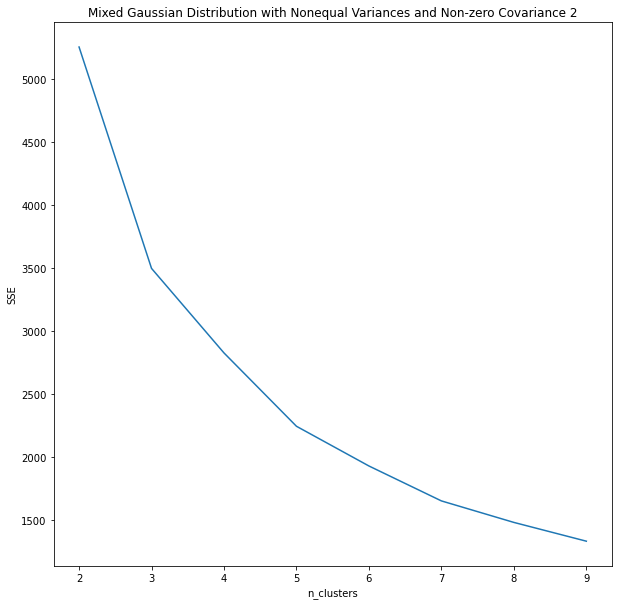
**Data 8:**

1. 数据可视化



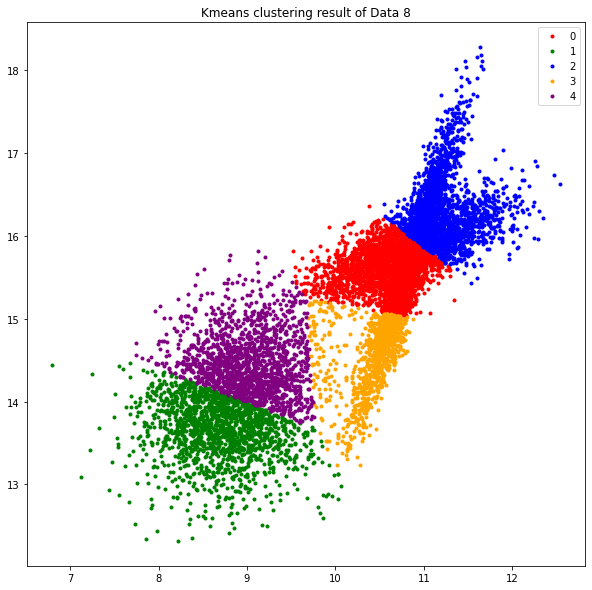
这次增大了重叠程度

2. 碎石图



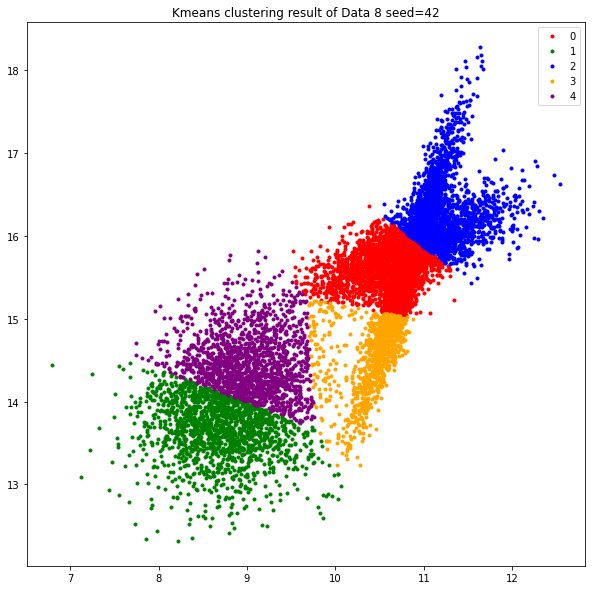
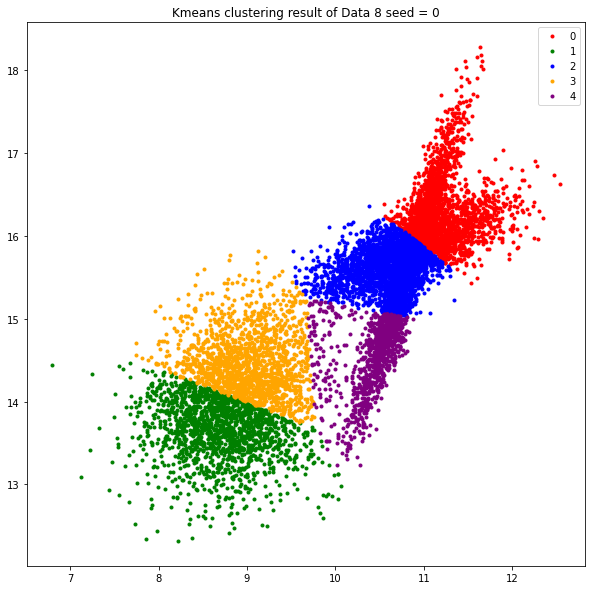
选取k = 3。

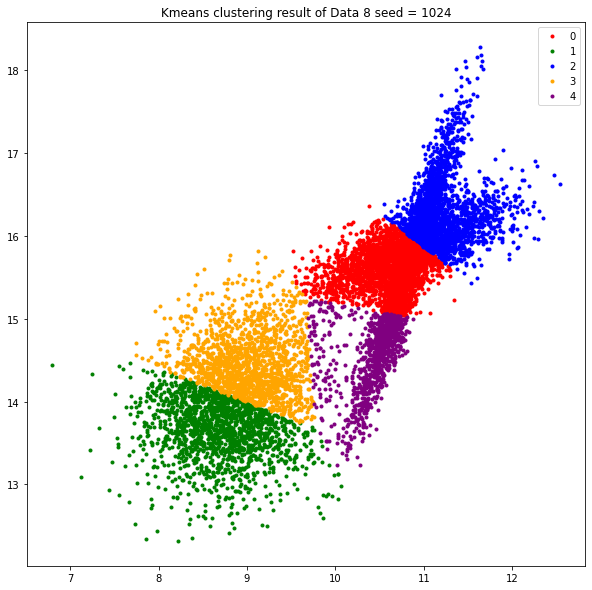
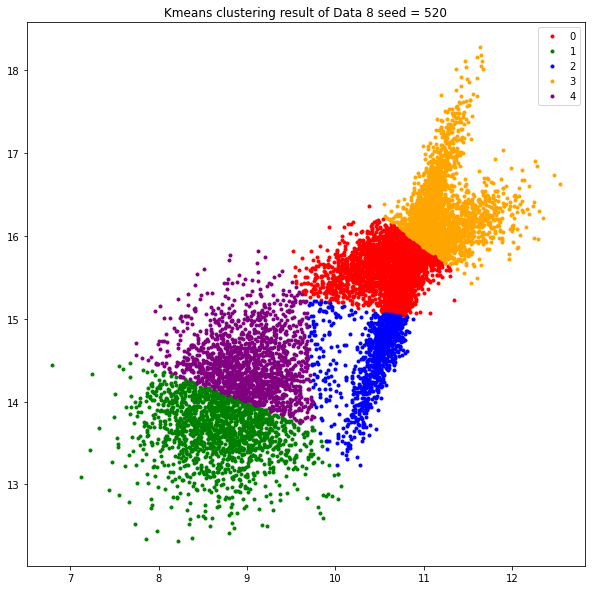
3. K-Means聚类结果

****

可见，kmeans非常迷茫

**二、 在Data 8 的基础上探究初始化对聚类结果的影响。**





可见，聚类结果是受随机初始化影响很小的，这说明kmeans迭代到最后是收敛的。但是聚类结果的编号受随机初始化影响较大，这是很容易理解的，因为kmeans算法的第一步就是随机选取初始点。