

**《机器学习》课程论文**

**[](http://bysj.zjgsu.edu.cn/)**

题目：基于聚类算法的客户个性分析

**学 院：统计与数学学院**

**专 业：应用统计2202**

**学 号：22020040149**

**学生姓名：王振宽**

**二○二三 年 六 月**

基于聚类算法的客户个性分析

摘要

随着大数据时代的到来，个性化营销越来越受到企业的关注。客户个性分析是一种重要的市场营销策略，通过对客户数据的聚类分析，可以将客户划分为不同的群体，从而实现精细化的营销推送。本论文选用了三种常用的聚类算法，包括K-means++、Agglomerative Clustering和Spectral Clustering，通过对一个数据集的实证研究，探讨了不同聚类算法在客户个性分析中的应用。

首先，本文介绍了K-means++、Agglomerative Clustering和Spectral Clustering这三种聚类算法的基本原理和特点，并针对研究问题设定了最小簇数为2，最大簇数为8的实验设置。接着，本文选用了Silhouette score作为聚类算法的评价指标，该指标可以衡量聚类结果的紧密性和分离性。通过在实验数据集上运行不同聚类算法，并计算其对应的Silhouette score，本文得出了最好的模型为Agglomerative Clustering，并且设置簇数为3。

进一步地，本文构建了特征的主要特征图，通过对聚类结果的可视化分析，得出了最终的分类结果。这对于企业进行客户个性化营销推送具有重要的实际应用意义，可以帮助企业更好地理解客户群体的特点和需求，从而优化营销策略，提高市场竞争力。

本研究的结果表明，Agglomerative Clustering算法在客户个性分析中具有较好的性能，可以有效地帮助企业进行客户细分和个性化营销推送。同时，特征的主要特征图的构建和可视化分析也为企业提供了直观的营销决策参考。这对于企业在面对大规模客户数据时，利用聚类算法进行个性化营销具有重要的实际应用价值。

**关键词：客户个性分析，K-means++，Agglomerative Clustering，Spectral Clustering，Silhouette score，特征可视化**

**目录**

[一、 问题的背景 4](#_Toc11860954)

[二、 数据集的介绍 5](#_Toc11860955)

2.1 People**…………………………………………………………………….**5

2.2 Products…………………………………………………………………..5

2.3 Promotion………………………………………………………………...6

[三、 介绍 7](#_Toc11860956)

[四、 模型的建立与求解 8](#_Toc11860957)

4.1 指标的选择……………………………………………………………...8

4.1.1 Silhouette coefficient………………………………………………8

4.1.2 Calinski-Harabasz Index…………………………………………..8

4.1.3 Davies-Bouldin Index……………………………………………..9

4.2 数据预处理……………………………………………………………..9

4.2.1 减少内存…………………………………………………………9

4.2.2 特征生成………………………………………………………..10

4.2.3 相关性分析……………………………………………………..11

4.2.4 数量变量的分布………………………………………………..11

4.2.5 数据清洗………………………………………………………..12

4.2.6 数据缩放………………………………………………………..13

4.2.7 降维……………………………………………………………..13

4.3 聚类……………………………………………………………………14

4.3.1 K-means ++………………………………………………………14

4.3.2 Aglomerative clustering………………………………………….18

4.3.3 Spectral clustering………………………………………………..21

4.4 构建主要特征图……………………………………………………….23

4.5 聚类分析的结果……………………………………………………….25

[参考文献 25](#_Toc11860958)

1. 问题的背景

客户个性分析是对公司理想客户的详细分析。它可以帮助企业更好地了解其客户，并使他们更容易根据不同类型客户的特定需求、行为和关注点修改产品。

客户个性分析可帮助企业根据来自不同类型客户群的目标客户修改其产品。例如，公司可以分析哪个客户群最有可能购买该产品，然后仅在该特定细分市场上销售该产品，而不是花钱向公司数据库中的每个客户推销新产品。

这种分析可以帮助企业更好地了解客户需求、洞察市场趋势、优化营销策略、改进产品设计等，从而更好地满足客户需求、提升客户满意度，从而获得更好的业务成果。

研究背景如下：

1. 日益增长的市场竞争：在当今市场经济环境中，竞争日益激烈，企业需要更好地了解客户，深入挖掘客户的个性化需求，以便制定更加精准的营销策略，提高市场份额和竞争力。

2. 数字化和大数据时代的到来：随着信息技术的迅速发展和数字化、网络化的普及，企业可以方便地获得大量客户数据，如购物记录、浏览历史、社交媒体行为等。这些数据为客户个性分析提供了更丰富的数据基础，促使企业更深入地了解客户行为和偏好。

3. 强调用户体验和个性化服务：现代消费者对于产品和服务的要求越来越高，强调个性化和定制化。企业需要了解客户的个性化需求，提供更加精准的产品和服务，以提升用户体验，增加用户忠诚度。

4. 数据驱动的决策和营销：企业日益依赖数据驱动的决策和营销策略，客户个性分析为企业提供了更多的数据支持和决策依据，帮助企业做出更加明智和有效的决策，从而提高企业的运营效率和业务绩效。

因此，客户个性分析作为一种重要的市场研究和营销策略工具，能够帮助企业深入了解客户的需求和行为，从而更好地满足客户需求，提升业务竞争力。

1. 数据集的介绍

我们首先对使用到的数据集进行简单介绍。

数据集来源于Kaggle竞赛Customer Personality Analysis，数据集的提供者是Dr. Omar Romero-Hernandez。共计2240条样本，29个特征，主要分为People、Products、Promotion和Place四部分。

下面是关于数据集内容的介绍：

**2.1 People**

共包括ID、Year\_Birth、Education、Marital\_Status、Income、Kidhome、Teenhome、Dt\_Customer、Recency和Complain等10个特征，具体类别见下表：

**表1： People类特征及其含义**

|  |  |
| --- | --- |
| 特征名 | 含义 |
| ID | 客户的唯一标识符 |
| Year\_Birth | 客户的出生年份 |
| Education | 客户的教育水平 |
| Marital\_Status | 客户婚姻状况 |
| Income | 客户的家庭年收入 |
| Kidhome | 客户家庭中的孩子数量 |
| Teenhome | 客户家庭中的青少年人数 |
| Dt\_Customer | 客户在公司注册的日期 |
| Recency | 自客户上次购买以来的天数 |
| Complain | 如果客户在过去 2 年内投诉过，则为 1，否则为 0 |

**2.2 Products**

共包括MntWines、MntFruits、MntMeatProducts、MntFishProducts、MntSweetProducts和MntGoldProds等六个特征。

**表2：Products类特征及其含义**

|  |  |
| --- | --- |
| 特征名 | 含义 |
| MntWines | 过去 2 年在葡萄酒上的花费 |
| MntFruits | 近两年水果消费金额 |
| MntMeatProducts | 过去 2 年在肉类上的花费 |
| MntFishProducts | 过去 2 年在鱼上的花费 |
| MntSweetProducts | 近两年在糖果上花费的金额 |
| MntGoldProds | 过去 2 年在黄金上花费的金额 |

**2.3 Promotion**

共包括NumDealsPurchases、AcceptedCmp1、AcceptedCmp2、AcceptedCmp3、AcceptedCmp4、AcceptedCmp5和Response等七个特征。

**表3：Promotion类特征及其含义**

|  |  |
| --- | --- |
| 特征名 | 含义 |
| NumDealsPurchases | 折扣购买次数 |
| AcceptedCmp1 | 如果客户在第一个活动中接受了报价，则为 1，否则为 0 |
| AcceptedCmp2 | 如果客户在第二次活动中接受了报价，则为 1，否则为 0 |
| AcceptedCmp3 | 如果客户在第三次活动中接受了报价，则为 1，否则为 0 |
| AcceptedCmp4 | 如果客户在第四次活动中接受了报价，则为 1，否则为 0 |
| AcceptedCmp5 | 如果客户在第五次活动中接受了报价，则为 1，否则为 0 |
| Response | 如果客户在上次活动中接受了报价，则为 1，否则为 0 |

**2.4 Place**

共包括NumWebPurchases、NumCatalogPurchases、NumStorePurchases、NumWebVisitsMonth等四个特征。

**表4：Place类特征及其含义**

|  |  |
| --- | --- |
| 特征名 | 含义 |
| NumWebPurchases | 通过公司网站进行的购买数量 |
| NumCatalogPurchases | 使用目录进行的购买次数 |
| NumStorePurchases | 直接在商店购买的数量 |
| NumWebVisitsMonth | 最近一个月访问公司网站的次数 |

我们的目标是：需要进行聚类来汇总客户群。

1. 介绍

在这项工作中，我将对商店中的客户细分问题进行聚类分析。

聚类分析（Cluster Analysis）是一种常用的数据分析方法，用于将一组数据对象（样本、观测）按照其相似性或距离度量划分为不同的组（簇、类别），使得同一组内的对象之间相似度较高，而不同组之间的相似度较低。聚类分析通常用于无监督学习任务，即没有预先定义的目标变量或标签。

聚类分析的主要目标是通过对数据对象的相似性或距离进行度量，将相似的对象聚集到同一组，从而实现数据的自动分类和分组。聚类分析通常包含以下步骤：

1. 选择合适的相似性度量或距离度量：根据问题的需求和数据的特点，选择适当的相似性度量或距离度量方法，例如欧氏距离、曼哈顿距离、余弦相似度等，用于度量数据对象之间的相似性或距离。
2. 数据预处理：对原始数据进行预处理，例如数据清洗、特征选择、特征标准化等，以确保数据的质量和一致性，减少噪声和异常值的影响。
3. 聚类算法选择：选择合适的聚类算法，例如K均值聚类、层次聚类、DBSCAN、密度聚类等，根据数据的特点和问题的需求，选择适合的聚类算法。
4. 聚类过程：根据选择的聚类算法，对数据对象进行聚类操作，将相似的对象划分到同一组，形成不同的簇或类别。
5. 结果评估：对聚类结果进行评估，通常使用一些内部评价指标（如簇内紧密度、簇间离散度等）或外部评价指标（如与已知标签的比较等）来评估聚类的质量和合理性。
6. 结果解释和应用：根据聚类结果进行进一步的数据解释和应用，例如可视化展示聚类结果、对不同簇进行特征分析、进行业务决策等。

作为这项工作的一部分，将考虑用于聚类分析的机器学习的所有阶段。这项工作的目标是根据聚类分析的结果为业务获得必要的建议。

1. 模型的建立与求解
   1. **指标的选择**

聚类有多种指标，例如：

**4.1.1 Silhouette coefficient**

轮廓系数（Silhouette coefficient）是一种用于评估聚类结果质量的指标，它可以用来度量聚类算法对于数据对象的聚类紧密度和分离度。轮廓系数的取值范围在-1到1之间，数值越接近1表示聚类结果越好，数值越接近-1表示聚类结果越差，而数值接近0表示聚类结果的质量一般。

轮廓系数的计算方法如下：

1. 对于每个数据对象，计算它与同一簇内其他对象的平均距离，得到该对象的聚类紧密度。
2. 对于每个数据对象，计算它与其他簇的平均距离，得到该对象到其他簇的簇间分离度。
3. 计算每个数据对象的轮廓系数，即轮廓系数 = （簇间分离度 - 聚类紧密度）/ max(簇间分离度, 聚类紧密度)。
4. 计算所有数据对象的轮廓系数的平均值，作为整个聚类结果的轮廓系数。

轮廓系数越接近1，表示聚类结果越好，表示同一簇内的数据对象越相似，不同簇之间的数据对象越分离。轮廓系数越接近-1，表示聚类结果越差，表示同一簇内的数据对象差异较大，不同簇之间的数据对象较为相似。轮廓系数接近0，表示聚类结果的质量一般。

**4.1.2** **Calinski-Harabasz Index**

Calinski-Harabasz Index，也称为CH指数，是一种用于评估聚类结果质量的指标，它可以用来度量聚类算法对于数据对象的聚类紧密度和分离度。与轮廓系数类似，CH指数可以帮助判断聚类结果的好坏，越高的CH指数表示聚类结果越好。

CH指数的计算方法如下：

1. 对于每个簇，计算该簇内数据对象之间的平均距离，得到该簇的簇内紧密度。

2. 对于每一对不同的簇，计算这两个簇之间的平均距离，得到簇间分离度。

3. 计算每个簇的CH指数，即CH指数 = （簇间分离度 + 簇内紧密度）/ max(簇间分离度, 簇内紧密度)。

4. 计算所有簇的CH指数的平均值，作为整个聚类结果的CH指数。

**4.1.3 Davies-Bouldin Index**

Davies-Bouldin Index，简称DBI，是一种用于评估聚类结果质量的指标，用于度量聚类算法对于数据对象的聚类紧密度和分离度。

DBI的计算方法如下：

1. 对于每个簇，计算该簇内数据对象之间的平均距离，得到该簇的簇内紧密度。

2. 对于每一对不同的簇，计算这两个簇之间的距离度量，例如欧氏距离、曼哈顿距离等，得到簇间分离度。

3. 计算每个簇的DBI，即DBI = （簇间分离度 + 簇内紧密度）/ max(簇间分离度, 簇内紧密度)。

4. 计算所有簇的DBI的平均值，作为整个聚类结果的DBI。

DBI越小，表示聚类结果越好，表示同一簇内的数据对象越相似，不同簇之间的数据对象越分离。DBI可以作为一种评估聚类结果质量的指标，可以用于比较不同聚类算法的性能，选择最优的聚类算法，或者用于调整聚类算法的参数，以优化聚类结果。

在本文的工作中，为了评估聚类质量，我们将使用 Silhouette 系数度量以及降维后，我们将查看三维空间中的数据，以直观评估聚类质量。

**4.2 数据预处理**

**4.2.1 减少内存**

我们在保持数据准确性的前提下，减少DataFrame的内存占用，从而提高数据处理效率和降低内存消耗。

具体而言，算法主要步骤如下：

1. 计算输入DataFrame的初始内存占用，单位为MB，并输出提示信息。

2. 遍历DataFrame的每一列，获取列的数据类型。

3. 根据列的数据类型，判断是否为整数类型（int），如果是，则判断列的最小值和最大值是否在对应整数数据类型的取值范围内（例如np.int8、np.int16、np.int32、np.int64），如果是，则将列的数据类型转换为相应的整数类型，从而减少内存占用。

4. 如果列的数据类型不是整数类型，再判断是否为浮点数类型（float），如果是，则判断列的最小值和最大值是否在对应浮点数数据类型的取值范围内（例如np.float32），如果是，则将列的数据类型转换为相应的浮点数类型，从而减少内存占用。

5. 如果列的数据类型不是整数类型或浮点数类型，将其转换为Pandas的分类（category）数据类型，从而进一步减少内存占用。

6. 计算优化后的DataFrame内存占用，并输出优化后的内存占用和内存减少比例的提示信息。

7. 返回优化后的DataFrame对象。

我们最初的DataFrame的内存为0.5MB，经过优化后的内存为0.11MB，内存下降了78%。

**4.2.2 特征生成**

特征生成是指通过对原始数据进行一系列处理和转换，从中提取出新的特征变量，以用于机器学习、数据挖掘等任务。特征生成的目的是为了改善模型的性能，提高预测准确性和模型的解释能力。

1. Age

通过从当前年份中减去客户的出生年份，从而得到客户的年龄。这可以将年份差作为一个新的特征添加到数据框中，用于后续的数据分析、建模等任务中。

2. Years\_Since\_Registration

自客户注册后的年数，这个特征可以帮助分析客户的忠诚度、活跃度以及对公司产品和服务的长期使用情况等。

3. Family\_Size

Family\_Size列中的值将表示家庭中的总人数，包括婚姻状态、Kidhome和Teenhome三个指标的综合信息，便于在后续的数据分析和建模中使用。

4. Sum\_Mnt

表示在多个产品上的总消费金额，是将MntWines、MntFruits、MntMeatProducts、MntFishProducts、MntSweetProducts和MntGoldProds汇总的结果。

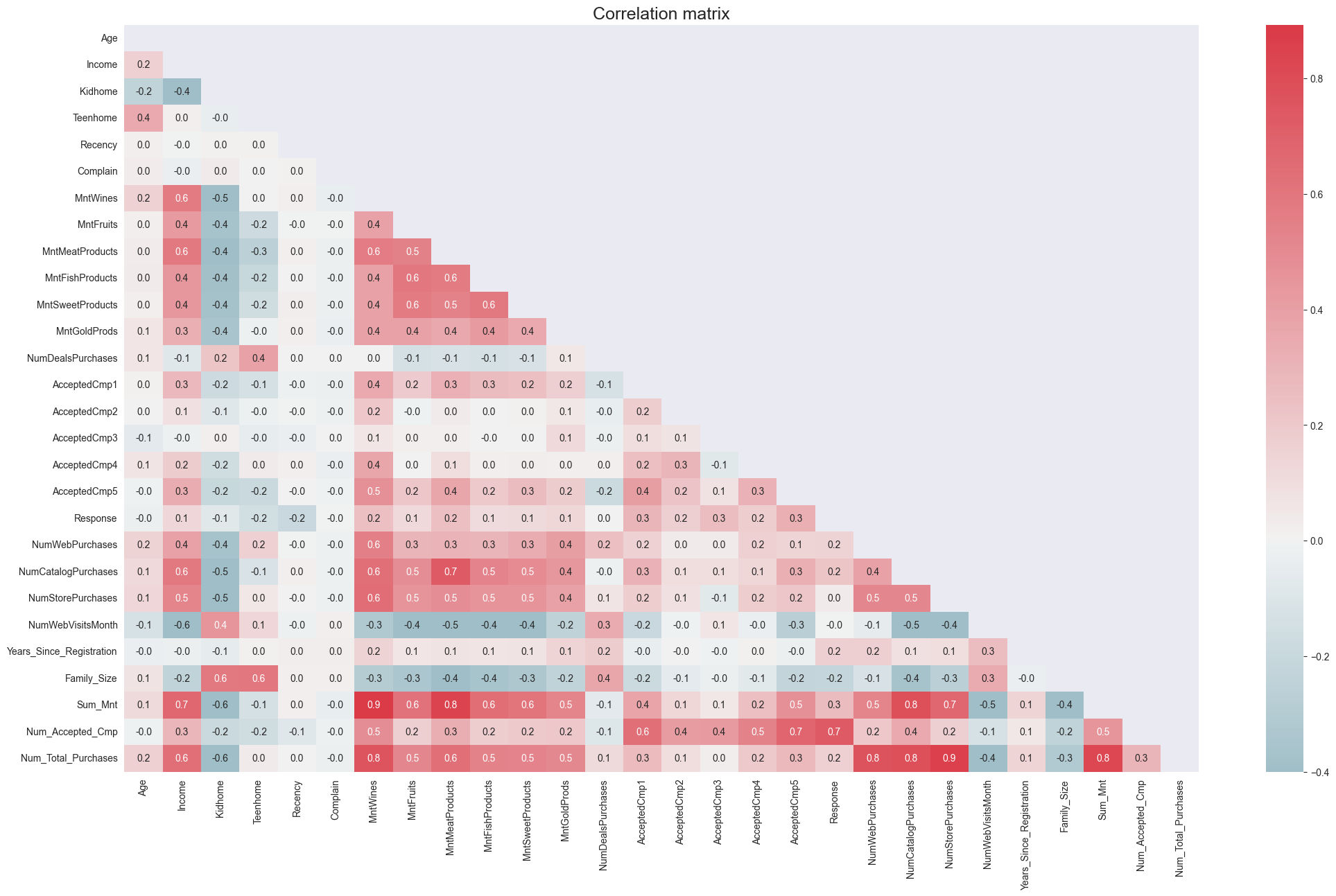
5.Num\_Accepted\_Cmp

表示客户在多少个公司中接受了优惠活动的邀请，是将AcceptedCmp1、AcceptedCmp2、AcceptedCmp3、AcceptedCmp4、AcceptedCmp5和Response汇总的结果。

6. Num\_Total\_Purchases

表示客户的总购买次数，是将NumWebPurchases、NumCatalogPurchases和NumStorePurchases汇总的结果。

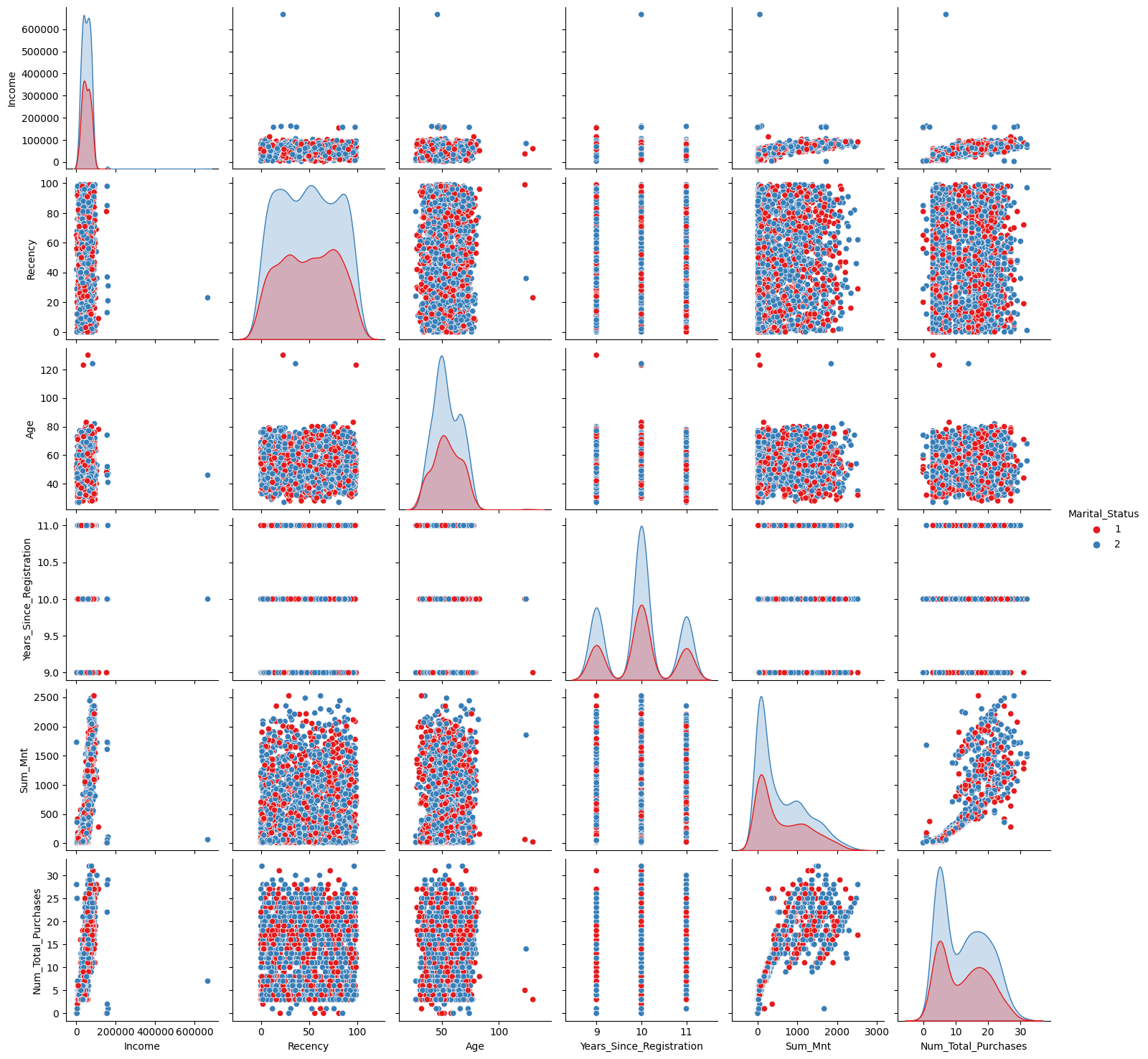
**4.2.3 相关性分析**



**图1：相关性热力图**

通过热力图可以直观地观察到不同列之间的相关性，颜色越深表示相关性越强。

**4.2.4 数量变量的分布**



**图2：数量变量的分布**

**4.2.5 数据清洗**

我们首先检查一下是否存在缺失值，整理后的表格如下：

**表4：缺失值情况**

|  |  |
| --- | --- |
| 特征名 | 缺失值数量 |
| ID | 0 |
| Year\_Birth | 0 |
| Education | 0 |
| Marital\_Status | 0 |
| Income | 24 |
| Kidhome | 0 |
| …… | ……. |
| Sum\_Mnt | 0 |
| Num\_Accepted\_Cmp | 0 |
| Num\_Total\_Purchases | 0 |

发现特征Income中有24条缺失值，因为样本数量较大，所以选择去除。

其次，我们决定去除数据中可能存在的异常值。处理步骤如下：

1. 计算了指定列的 25% 百分位数（q1）和 75% 百分位数（q3），并计算了两者之间的四分位距（IQR）；

2. 根据 IQR，计算了上界（upper\_bound）和下界（lower\_bound），其中上界定义为 q3 + 1.5  IQR，下界定义为 q1 - 1.5  IQR；

3. 通过逻辑运算符 & 来筛选出指定列中值在上界和下界之间的数据，从而移除了超出范围的异常值。

我们去除Age和Income两列特征中的异常值。

最后，我们删除无用的特征，Year\_Birth、ID、Z\_CostContact、Z\_Revenue、Dt\_Customer和Marital\_Status。

**4.2.6 数据缩放**

数据缩放对于使所有特征达到相同的比例是必要的。 如果不这样做，那么算法的注意力将被吸引到包含大值的特征。在这项工作中，归一化将用作数据缩放（作为归一化的结果，所有特征都在 0 到 1 的范围内）

计算公式如下：



**4.2.7 降维**

降维问题用于以下情况：

1. 数据集中有很多特征，我们想减少它们的数量，留下尽可能多的信息；

2. 我们在数据集中有很多特征，我们希望将数据可视化（例如，在 3D 空间中）；

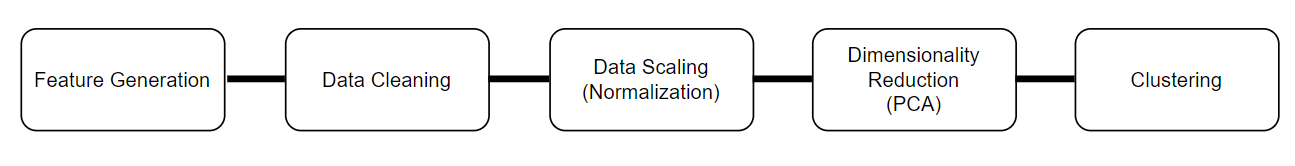
3. 解决多重共线性问题。

降维有很多不同的选择，它们基于不同的方法，例如：主成分分析 (PCA)、Uniform Manifold Approximation and Projection (UMAP)、t-distributed Stochastic Neighbor Embedding (t-SNE)、Locally-Linear Embedding (LLE)和Multidimensional Scaling (MDS)。

最常用的降维算法是 PCA，我们在这项工作中也使用了它。

本质上，在 PCA 中，我们从一个变量空间过渡到另一个变量空间，新空间包含更少的变量，其中新变量是不相关的，是旧变量的加权和。结果我们得到 m 个变量：{PC1, PC2, PC3...PCm}，其中 PC1 将收到最多的信息（最大样本方差），PC2 更少，依此类推。这项工作将使用 8 个主成分。

**4.3 聚类**



**图3：流程图**

在这项工作中，将使用以下聚类算法：

1. K-means ++

2. Aglomerative clustering

3. Spectral clustering

值得注意的是，在这些算法中，有必要预先指定集群的数量。我们假设最小簇数为二，最大簇数为八。基于此，针对不同的集群数量情况，我们将建立模型。如果具有大量集群的模型的度量是最大的，那么我们将迭代地增加集群的最大数量并构建新模型。

**4.3.1 K-means ++**

K-means++，仅从名字也可以看出它就是Kmeans聚类算法的改进版。它在初始化簇中心的方式上做了改进，其他地方同Kmeans聚类算法一样。

接下来是K-means++的原理：

1. 从数据集中随机（均匀分布）选取一个样本点作为第一个初始聚类中心；

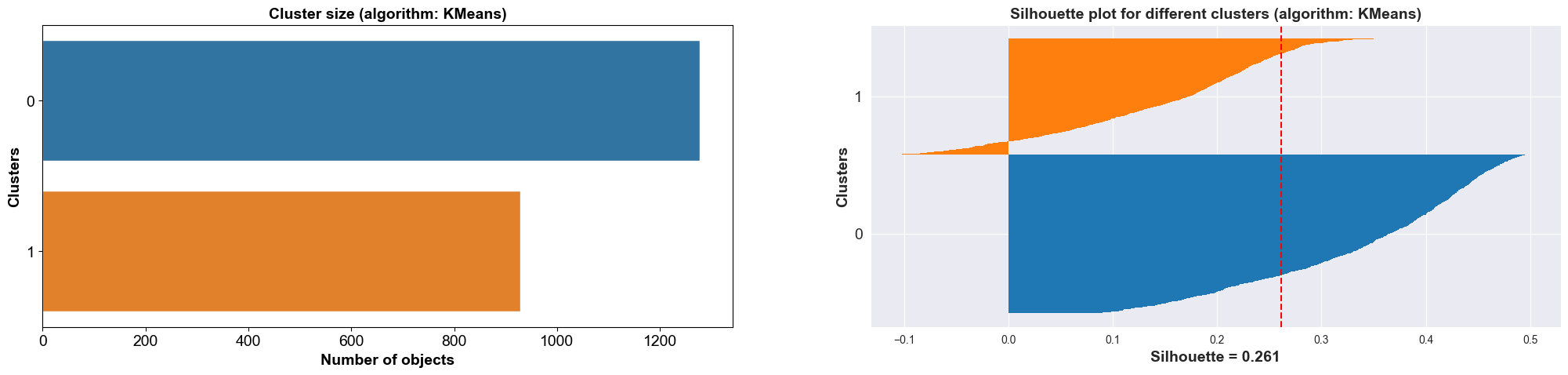
2. 接着计算每个样本与当前已有聚类中心之间的最短距离，用表示；然后计算每个样本点被选为下一个聚类中心的概率，最后选择最大概率值所对应的样本点作为下一个簇中心，其中：



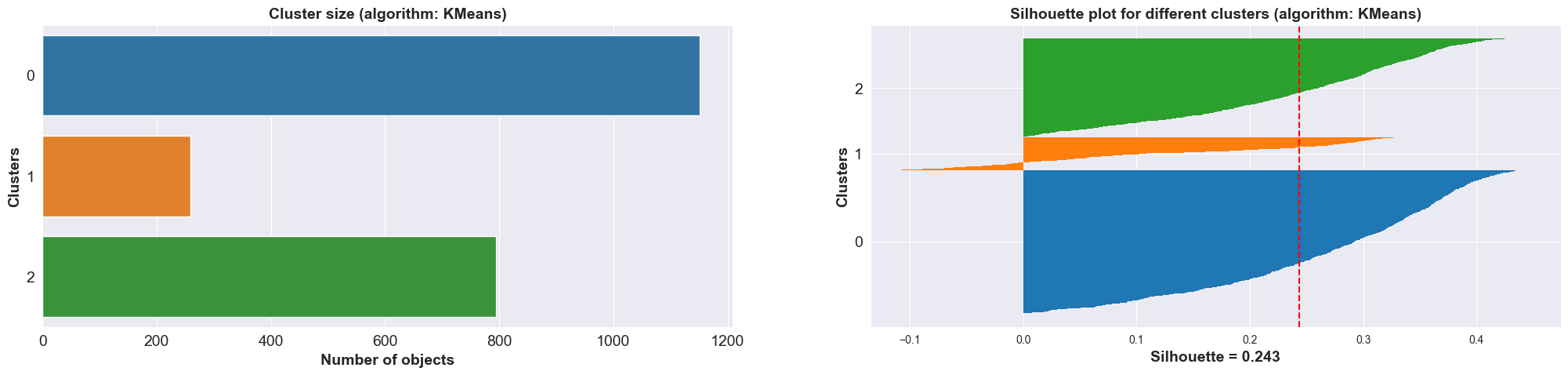
3. 重复第2步，直到选出个聚类中心。

从上述公式也可以看出，距离现有簇中心越远的样本点，越可能被选为下一个簇中心。

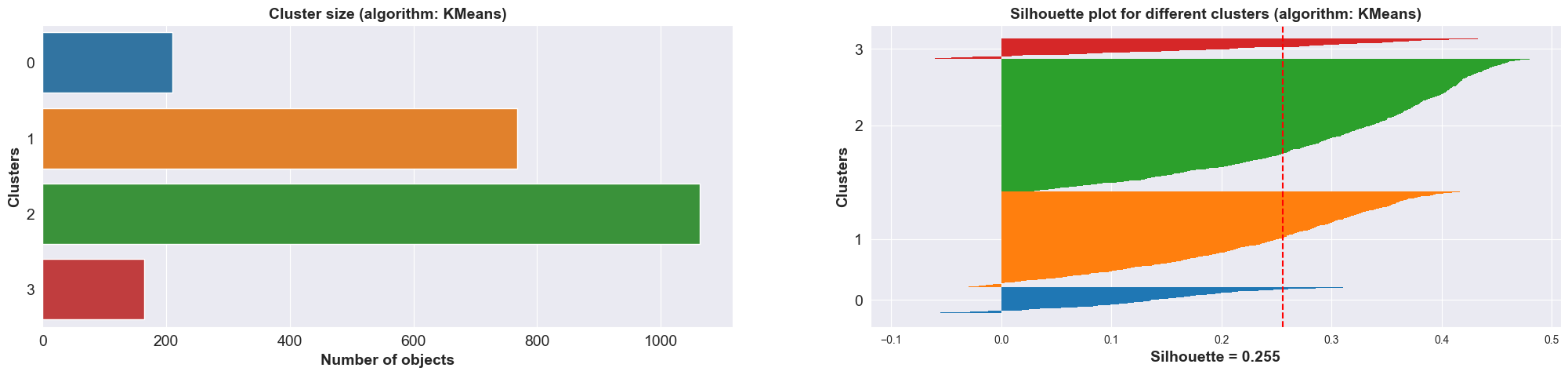
我们得到的结果如下：



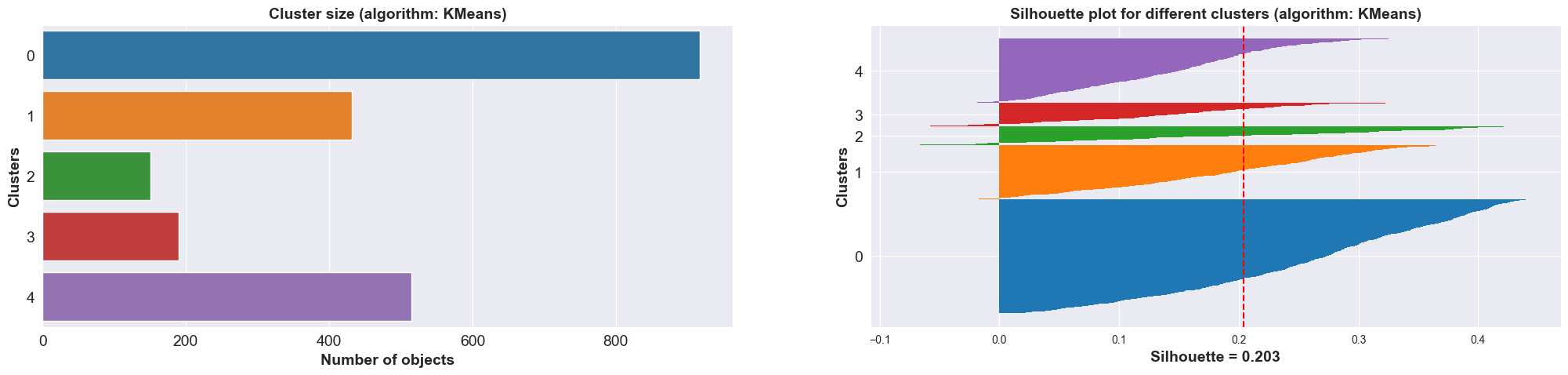
**图4：聚类中心为2**



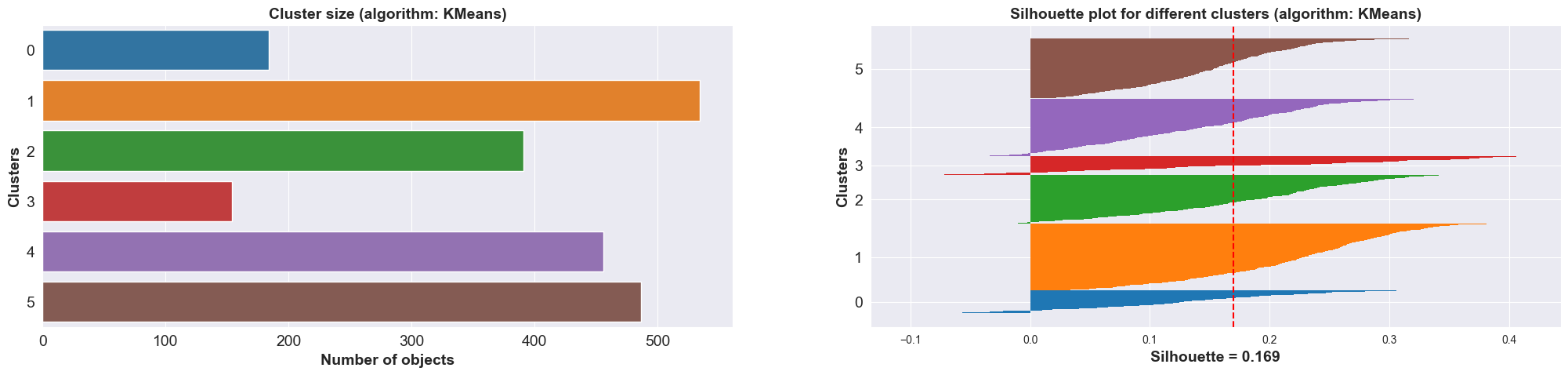
**图5：聚类中心为3**



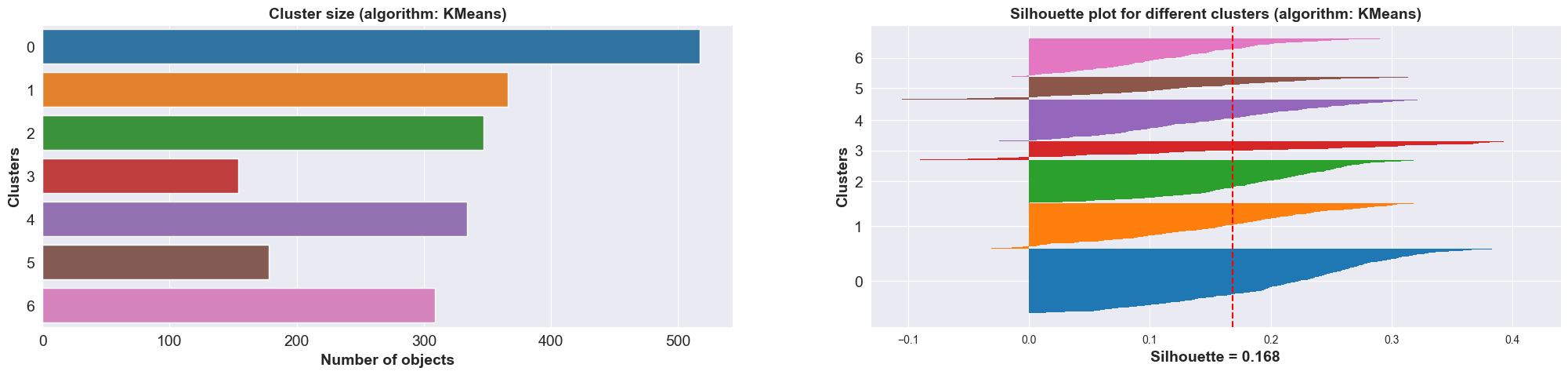
**图6：聚类中心为4**



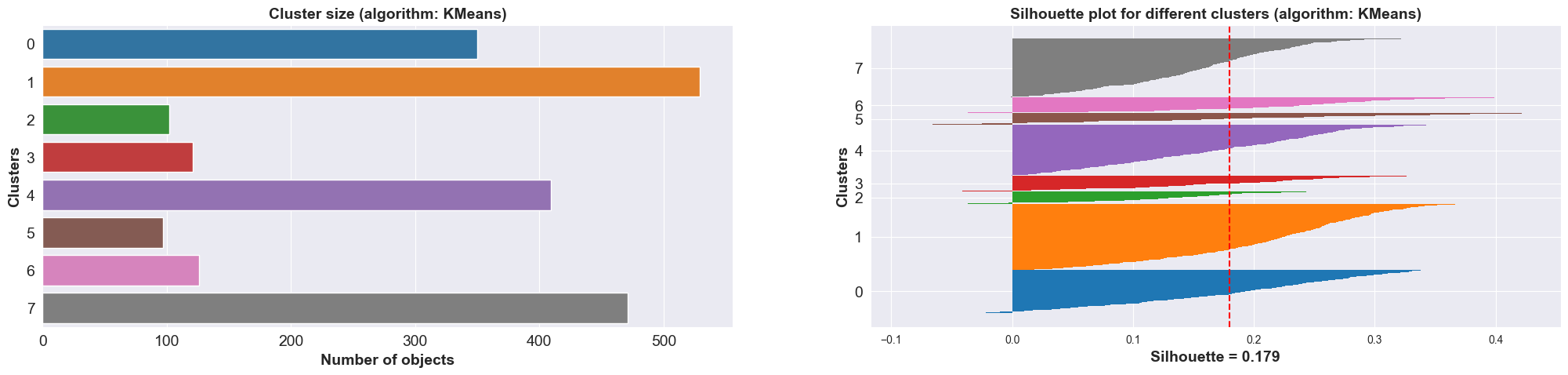
**图7：聚类中心为5**



**图8：聚类中心为6**



**图9：聚类中心为7**



**图10：聚类中心为8**

文中选择Silhouette score作为聚类算法性能的指标，用于衡量聚类结果的紧密度和分离度。计算方式如下：

1. 对于每个样本点，计算其与同簇内其他样本点的平均距离（称为a值），表示该样本点与簇内其他样本点的紧密度。

2. 对于每个样本点，计算其与最近邻不同簇内的所有样本点的平均距离（称为b值），表示该样本点与其他簇的分离度。

3. 计算每个样本点的轮廓系数（silhouette coefficient），定义为(b - a) / max(a, b)。轮廓系数的取值范围在-1到1之间，值越接近1表示样本点越紧密地聚类在一起且与其他簇分离得越好，值越接近-1表示样本点与其他簇更接近，值接近0表示样本点在簇内和其他簇之间的距离差异较小。

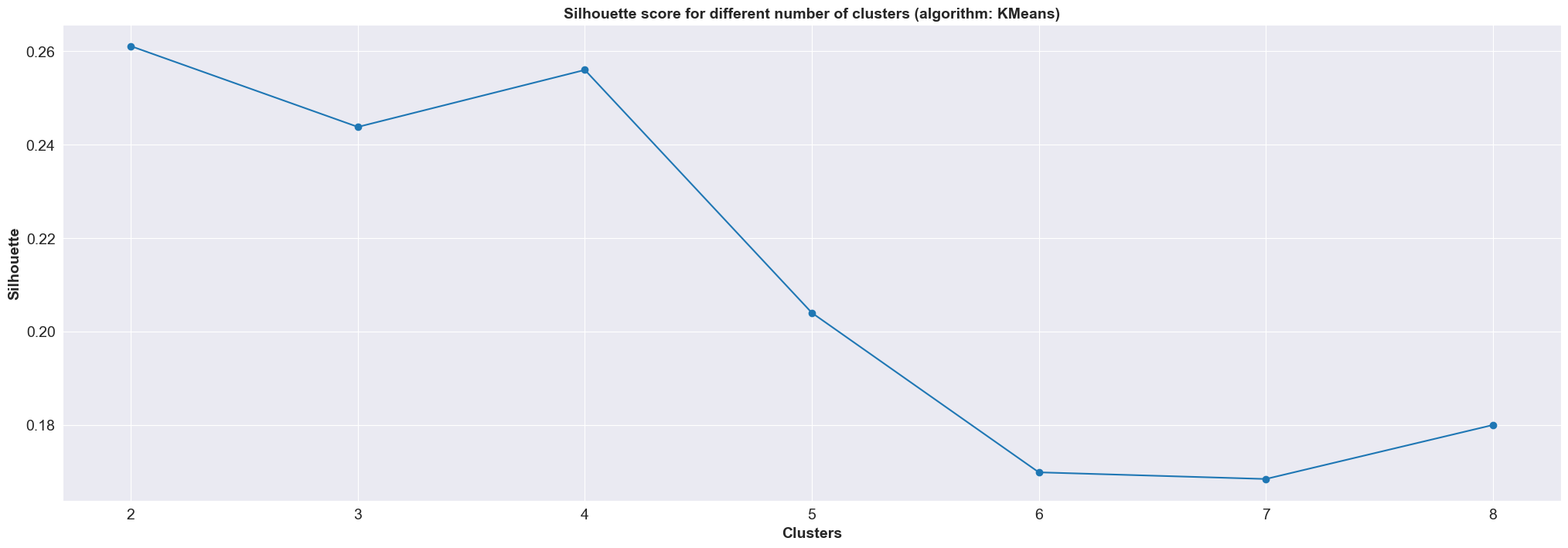
4. 对于所有样本点，计算轮廓系数的平均值，作为聚类结果的Silhouette score。

Silhouette score越接近1，表示聚类结果越好，样本点在同簇内紧密聚集且与其他簇分离较好；Silhouette score越接近-1，表示聚类结果较差，样本点与其他簇较接近；Silhouette score接近0，表示样本点在簇内和其他簇之间的距离差异较小，聚类结果可能存在一定的重叠。

我们的聚类中心从2个到8个，得到的silhouette值如下表所示：

**表6：K-means++聚类算法的silhouette值**

|  |  |
| --- | --- |
| 聚类中心 | Silhouette值 |
| 2 | 0.261 |
| 3 | 0.243 |
| 4 | 0.255 |
| 5 | 0.203 |
| 6 | 0.169 |
| 7 | 0.168 |
| 8 | 0.179 |



**图11：silhouette值随聚类中心的变化**

对于聚类算法而言，SSE（Sum of Squared Errors）也是一种常用的评价指标，用于衡量聚类结果的离散度或者聚类内部的差异性，从而评估聚类算法的聚类性能。

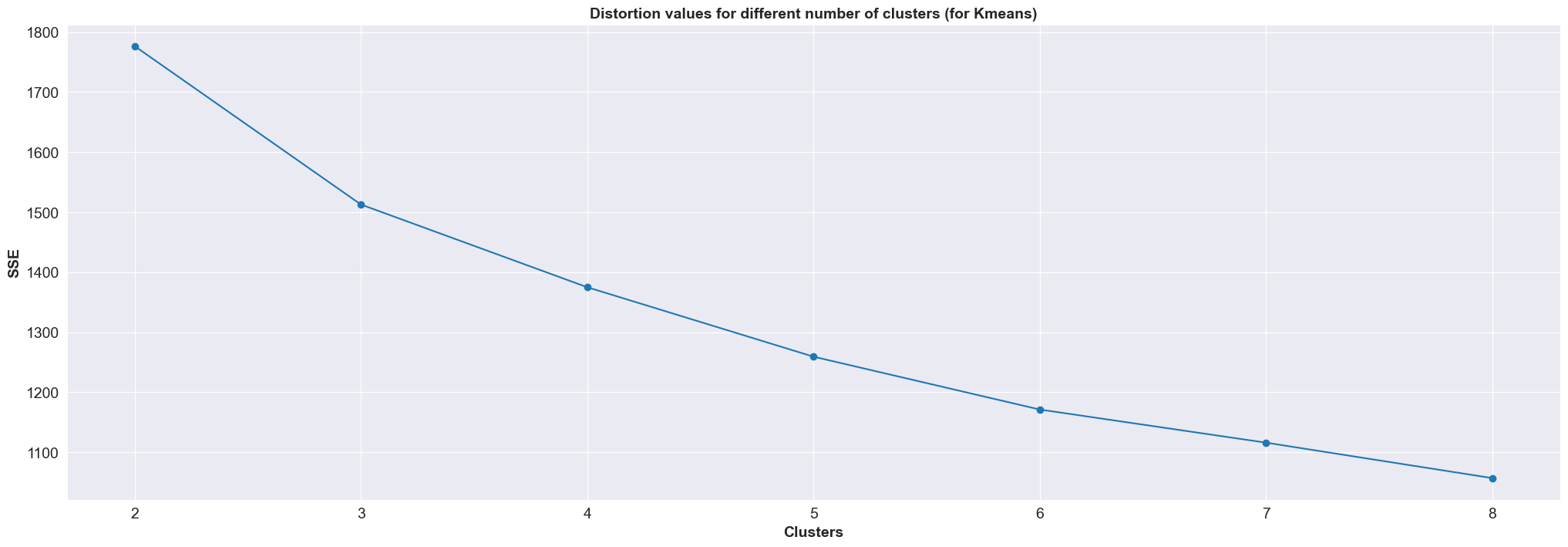
SSE的计算方式如下：

1. 对于每个聚类簇，计算其中所有样本点与该簇的质心（或者其他代表性样本点）之间的欧氏距离的平方。

2. 将每个聚类簇中所有样本点与质心之间的距离平方进行求和，得到该聚类簇的SSE。

3. 对于所有聚类簇，将其SSE相加，得到总的SSE。

我们得到不同聚类中心对应的SSE值如下：



**图12：SSE的值随聚类中心的变化**

SSE的值越小，表示聚类结果中簇内样本点之间的距离越近，即簇内的样本点越相似，簇内的差异性越小，聚类结果越好。因此，SSE越小越好。

**4.3.2 Aglomerative clustering**

AgglomerativeClustering是一种层次聚类（Hierarchical Clustering）算法，用于将数据点聚类成不同的群组。层次聚类是一种自底向上的聚类方法，通过逐步合并相邻的数据点或群组，形成一个层次化的聚类结构。下面大致讲一下 AgglomerativeClustering算法。

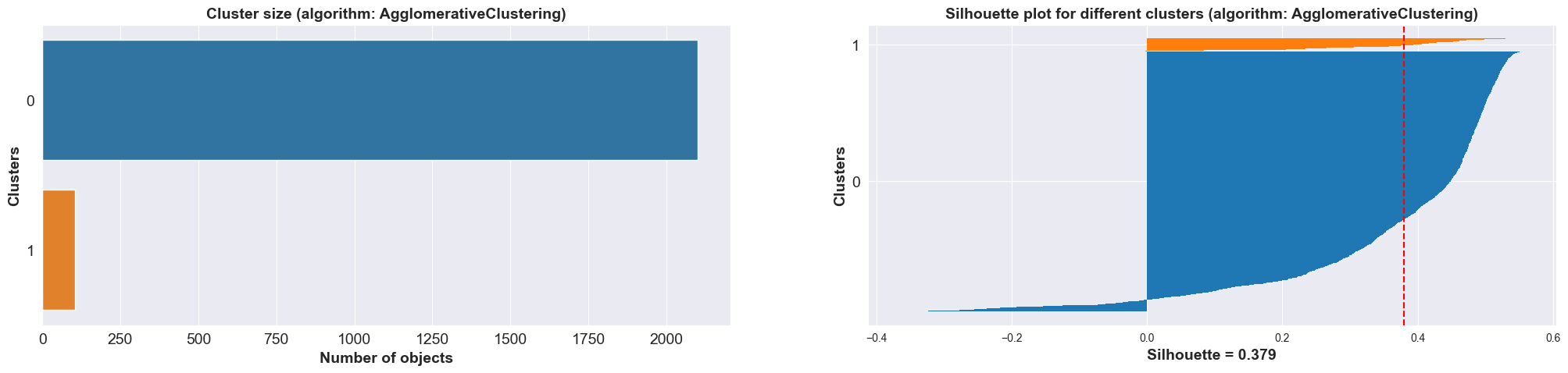
算法的原理很简单，最开始的时候将所有数据点本身作为簇，然后找出距离最近的两个簇将它们合为一个，不断重复以上步骤直到达到预设的簇的个数。

1. 将每一个元素单独定为一类；

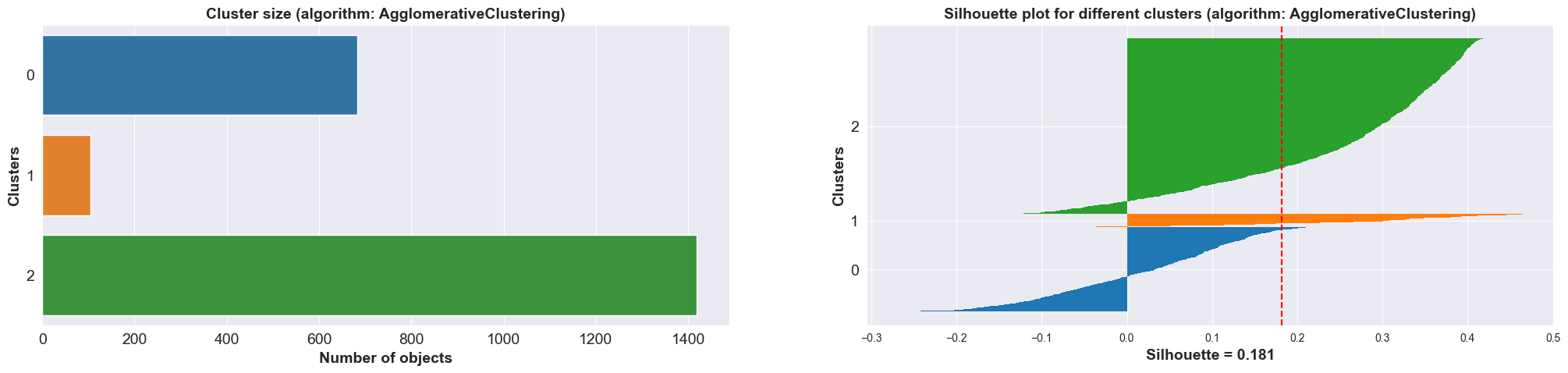
2. 每一轮都合并指定距离（对指定距离的理解很重要）最小的类；

3. 迭代第二步，直到所有的元素都归为同一类/类别数量已经达到了我们需要的数量。

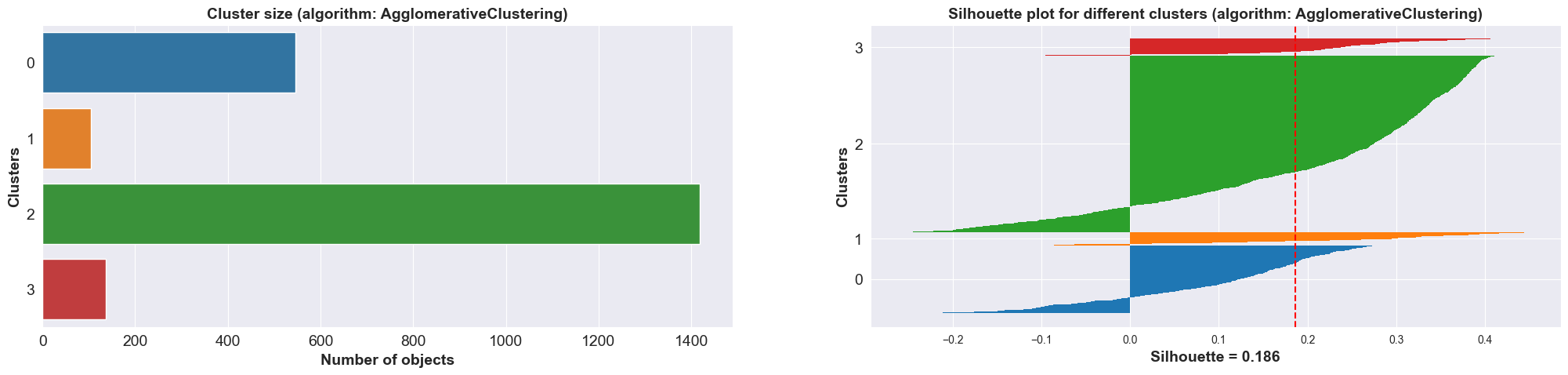
与上面的方法一致，我们分别选择不同的聚类中心，得到的结果如下：



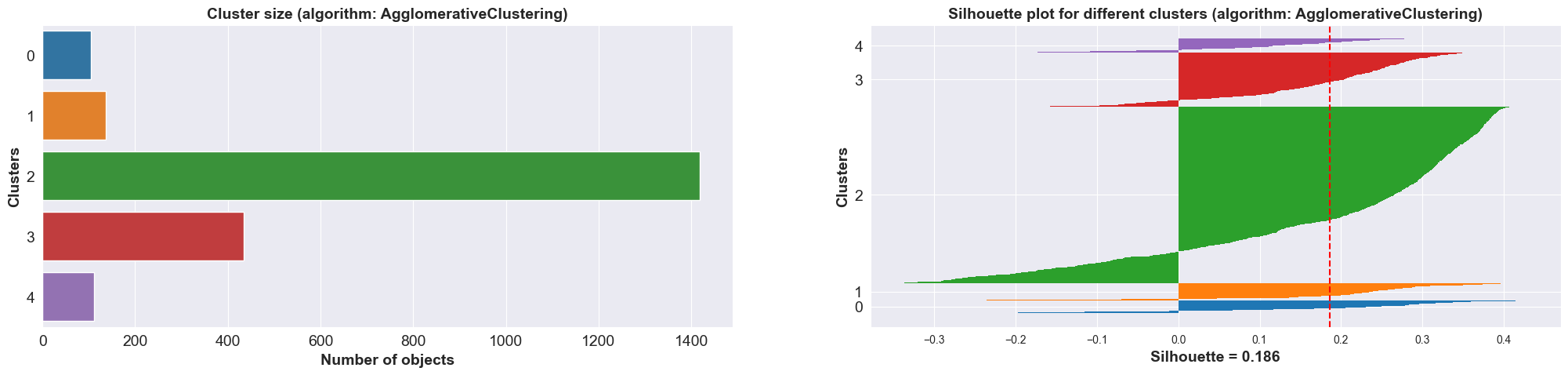
**图13：聚类中心为2**



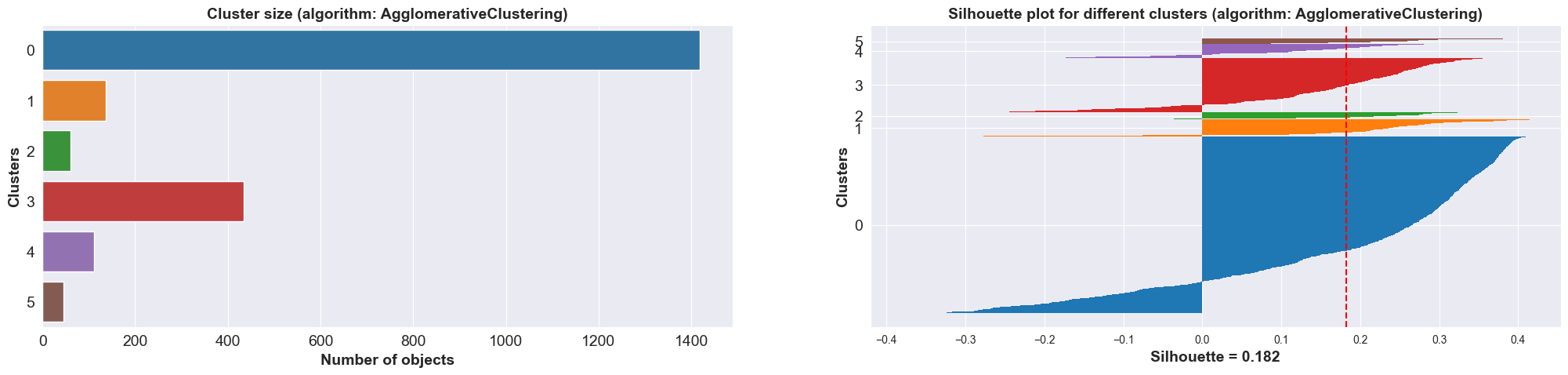
**图14：聚类中心为3**



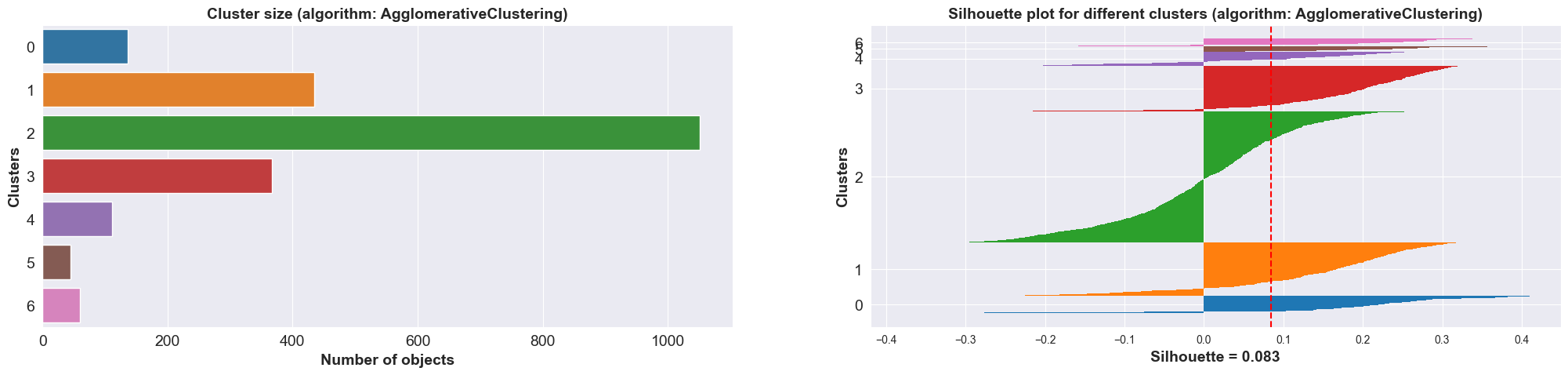
**图15：聚类中心为4**



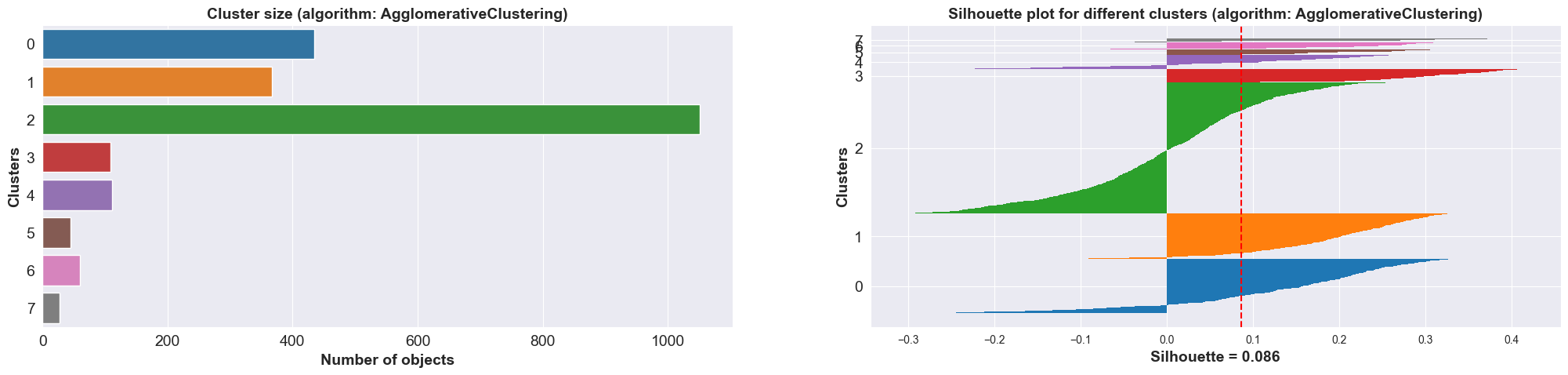
**图16：聚类中心为5**



**图17：聚类中心为6**



**图18：聚类中心为7**

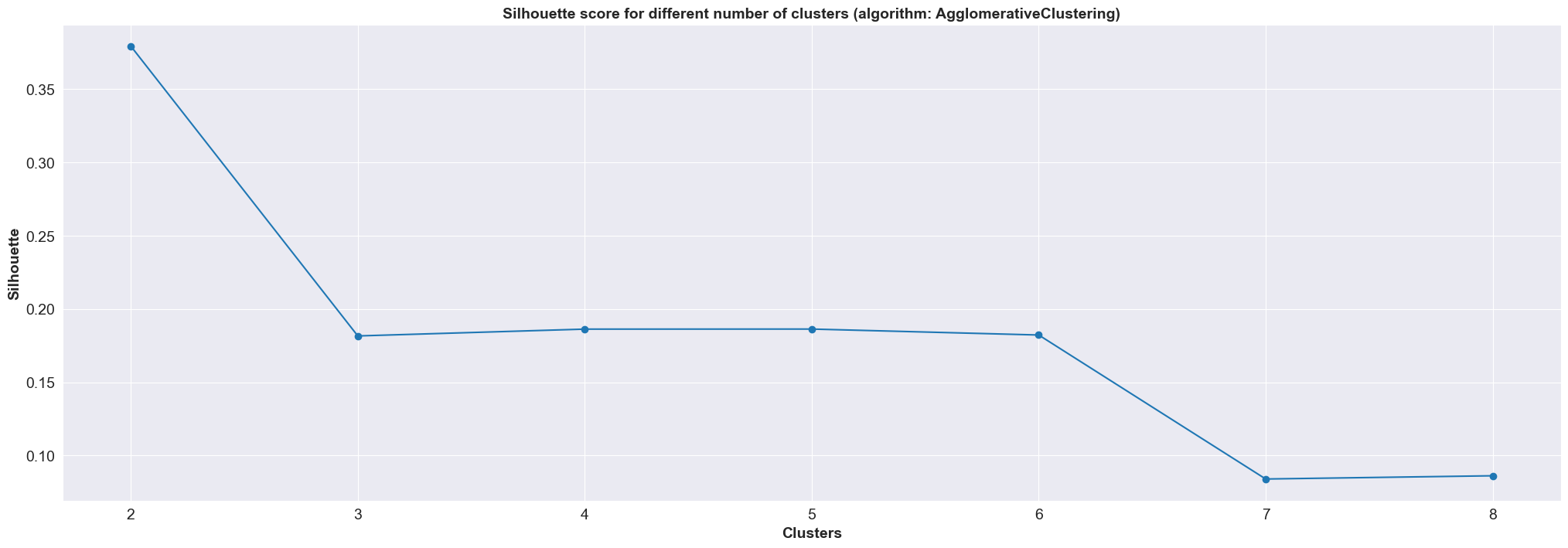


**图19：聚类中心为8**

我们的聚类中心从2个到8个，得到的Silhouette值如下表所示：

**表7：AgglomerativeClustering算法的Silhouette值**

|  |  |
| --- | --- |
| 聚类中心 | Silhouette值 |
| 2 | 0.379 |
| 3 | 0.181 |
| 4 | 0.186 |
| 5 | 0.186 |
| 6 | 0.182 |
| 7 | 0.083 |
| 8 | 0.086 |



**图20：silhouette值随聚类中心的变化**

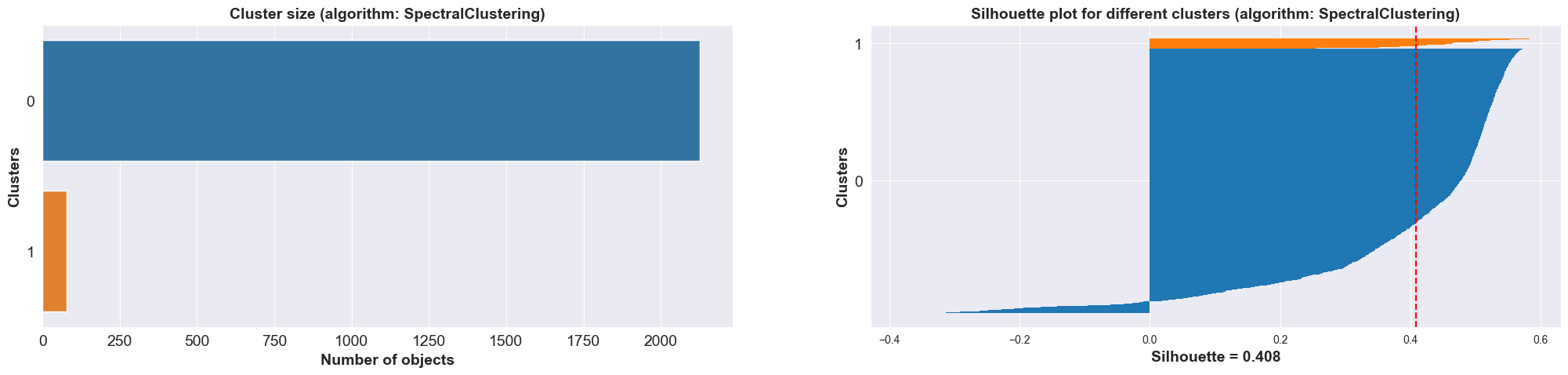
**4.3.3 Spectral clustering**

SpectralClustering是一种基于图谱理论的聚类算法，用于将数据点聚类成不同的群组。该算法通过构建数据点之间的相似度图谱，并利用图谱中的特征值和特征向量进行聚类。

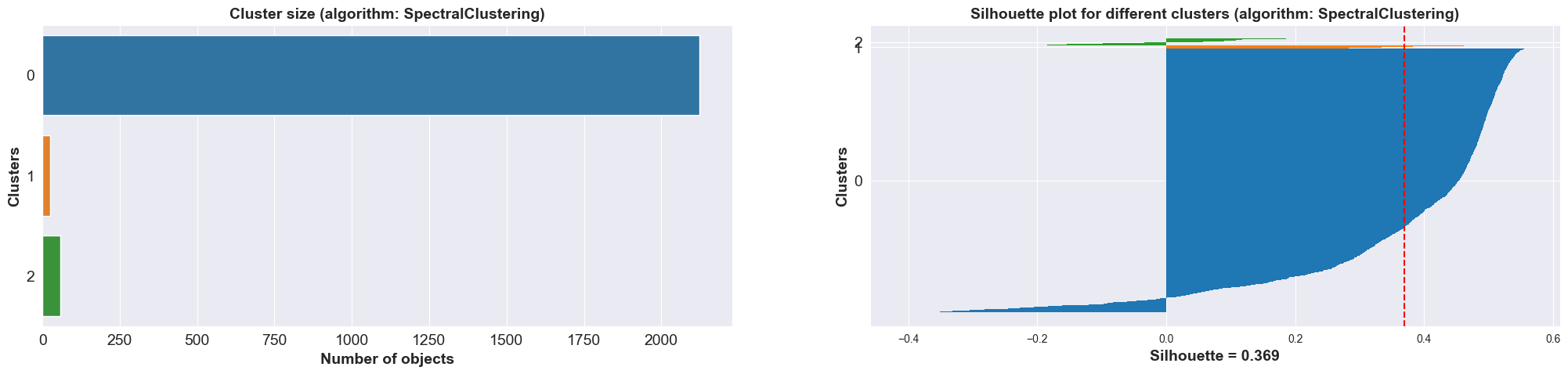
SpectralClustering算法基于以下思想：首先，将数据点看作图谱中的节点，并计算节点之间的相似度；然后，将相似度矩阵转换为拉普拉斯矩阵，并计算拉普拉斯矩阵的特征值和特征向量；最后，通过对特征向量进行聚类，将数据点划分为不同的簇。

SpectralClustering算法具有较强的聚类能力，可以处理复杂的非线性数据分布。该算法对数据点之间的相似度的计算较为灵活，可以根据实际问题选择不同的相似度度量方法，例如高斯核函数、K近邻等。

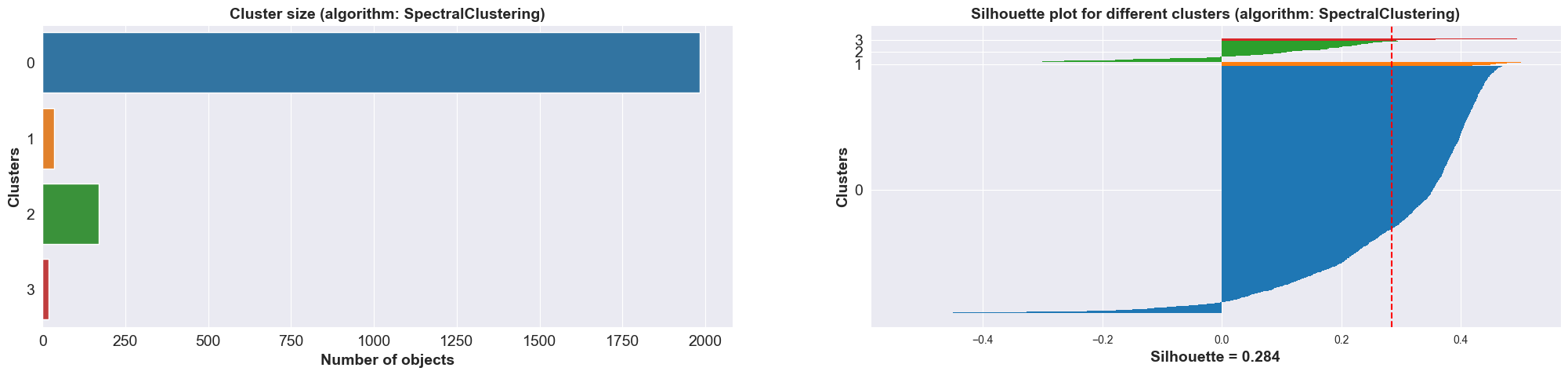
与上面的方法一致，我们分别选择不同的聚类中心，得到的结果如下：



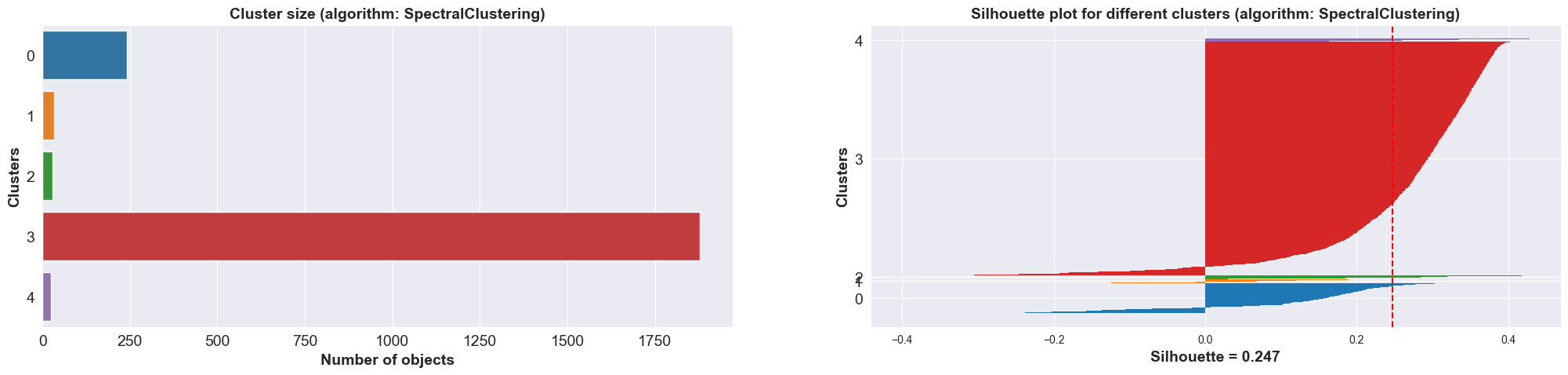
**图21：聚类中心为2**



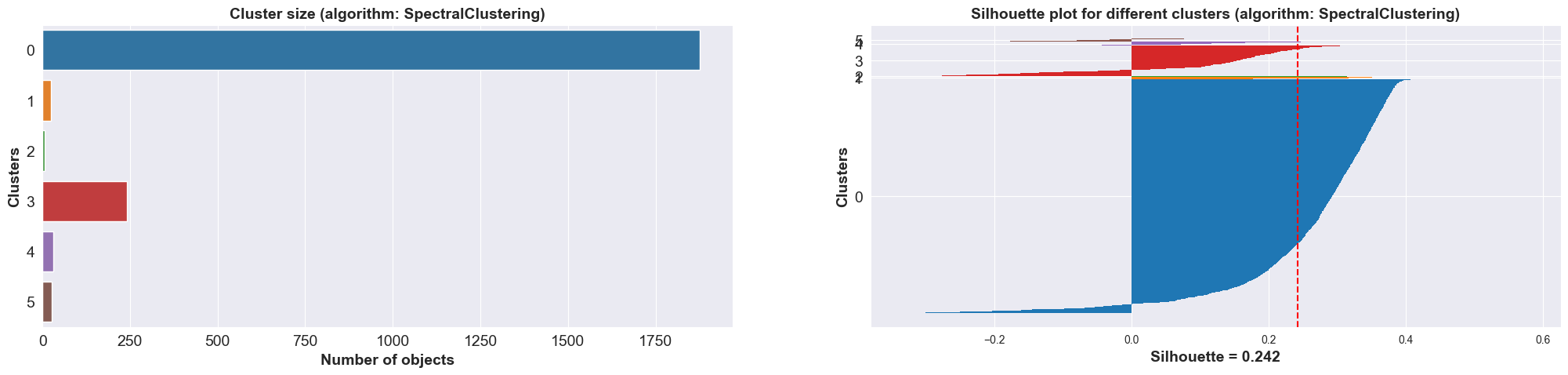
**图22：聚类中心为3**



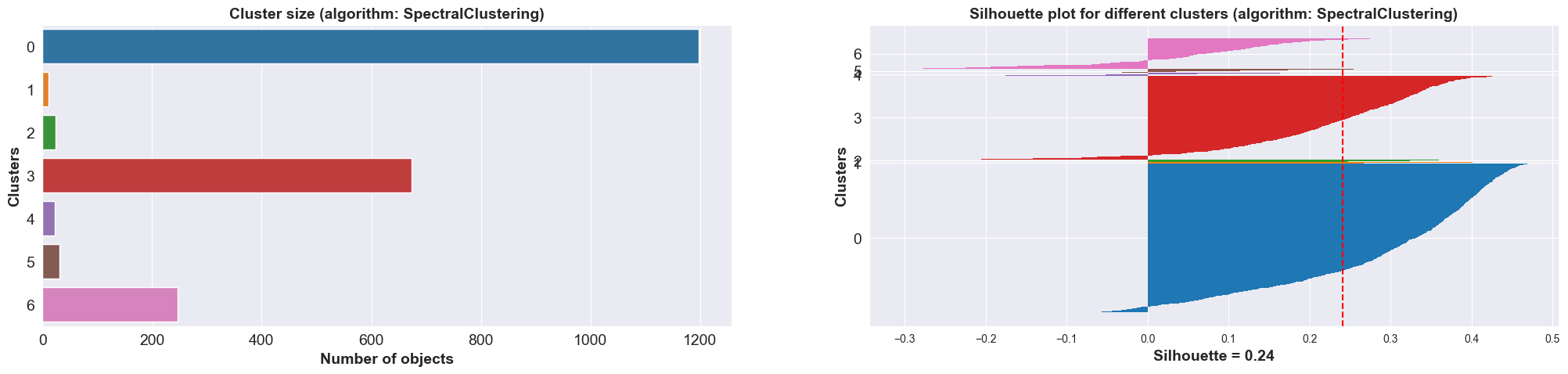
**图23：聚类中心为4**



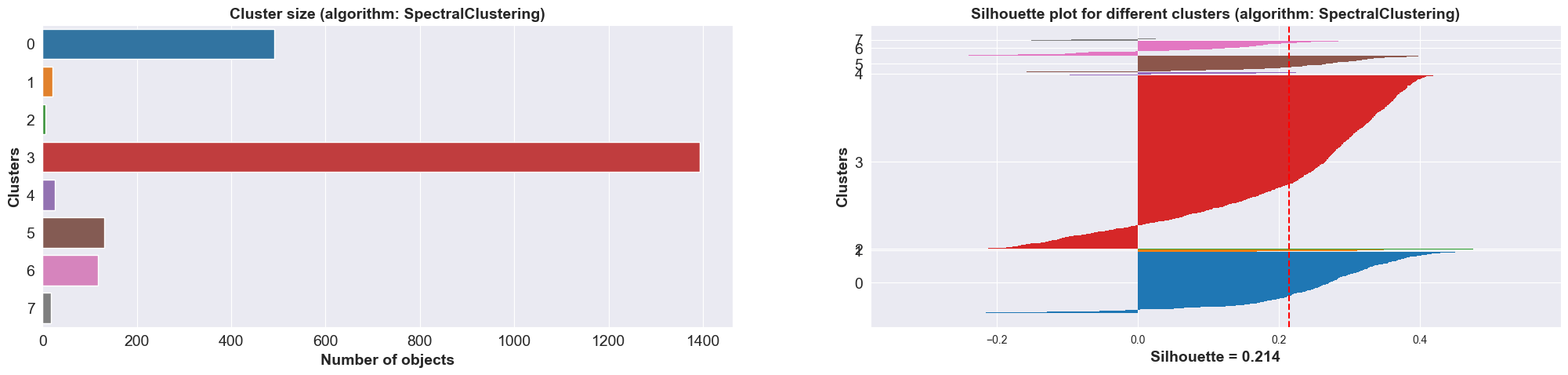
**图24：聚类中心为5**



**图25：聚类中心为6**



**图26：聚类中心为7**

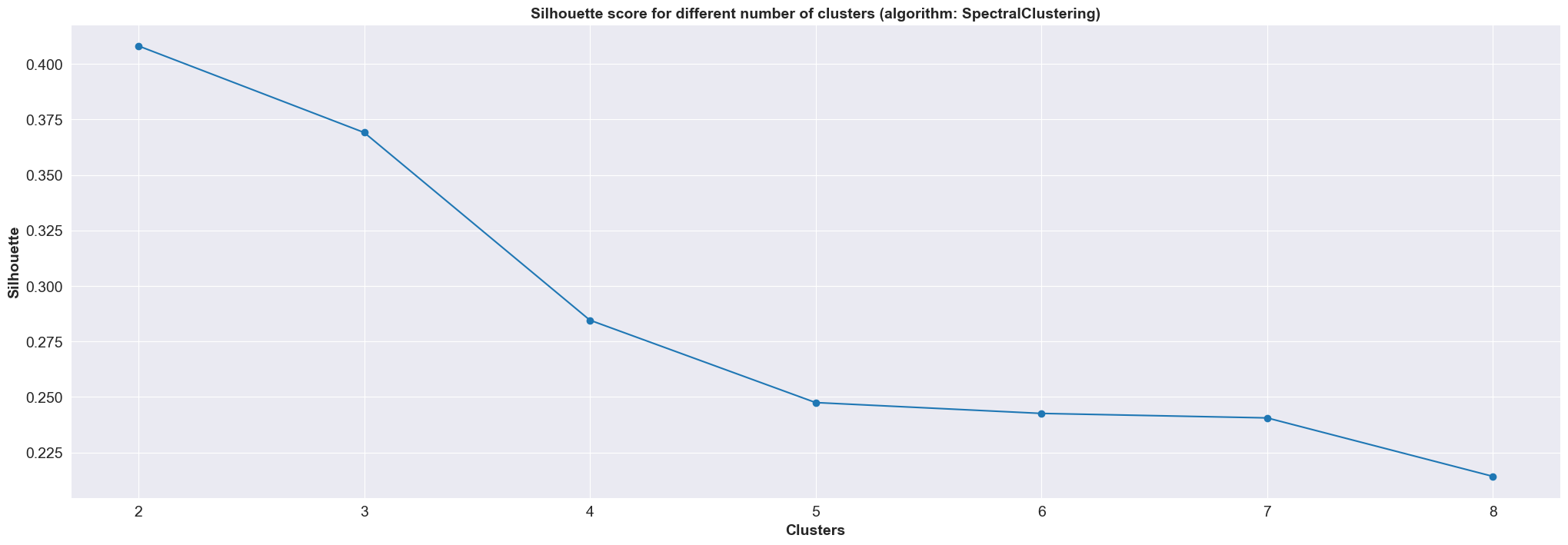


**图27：聚类中心为8**

我们的聚类中心从2个到8个，得到的Silhouette值如下表所示：

**表8：SpectralClustering算法的Silhouette值**

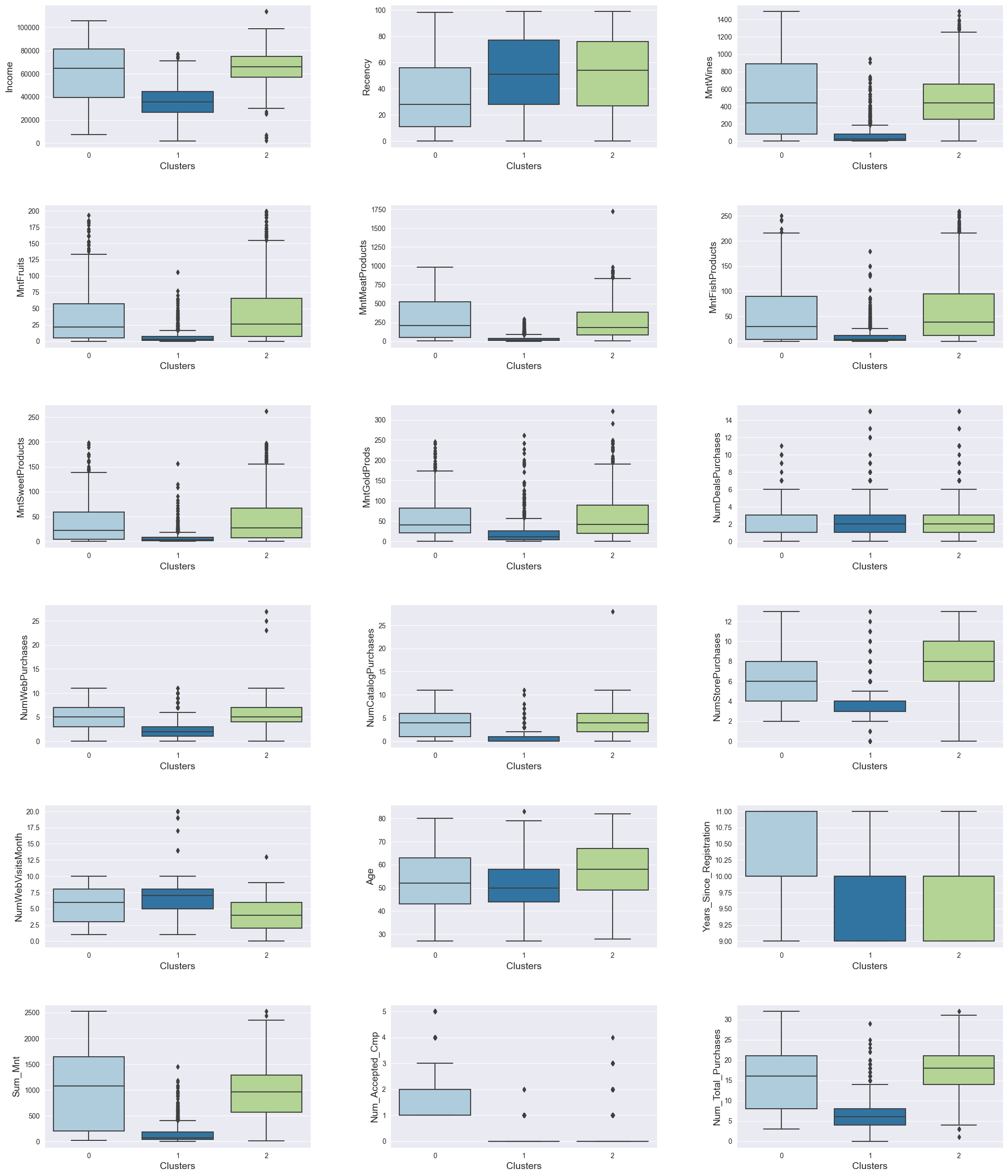
|  |  |
| --- | --- |
| 聚类中心 | Silhouette值 |
| 2 | 0.408 |
| 3 | 0.369 |
| 4 | 0.284 |
| 5 | 0.247 |
| 6 | 0.242 |
| 7 | 0.240 |
| 8 | 0.214 |



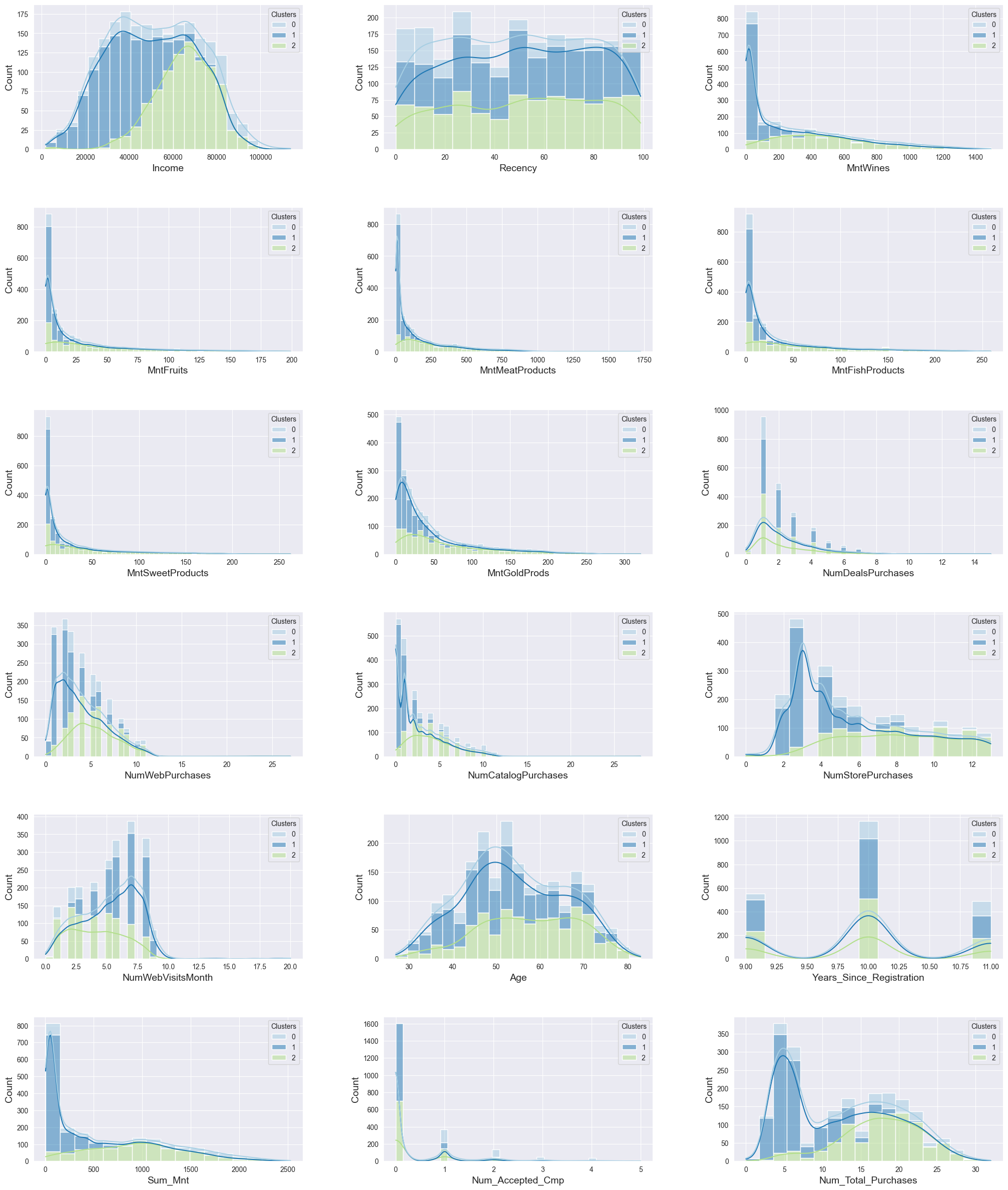
**图28：silhouette值随聚类中心的变化**

综上所述，我们得出最好的模型：AgglomerativeClustering(n\_clusters=3)。

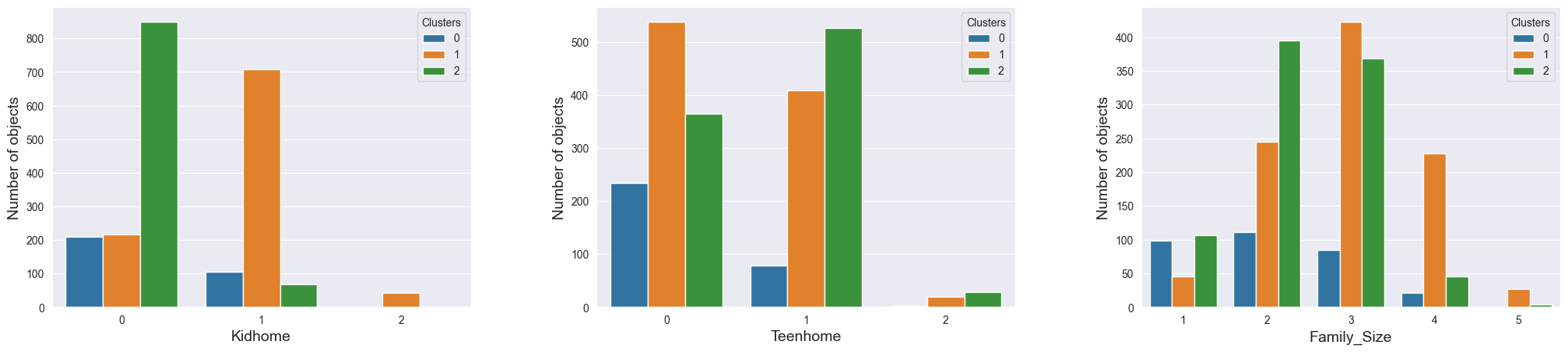
**4.4 构建特征的主要特征图**



**图29：主要特征分组后的箱型图**



**图30：主要特征分组后的分布图**



**图31：主要特征分组后的直方图**

**4.5 聚类分析的结果**

作为聚类分析的结果，我们将买家分为三组：

**4.5.1 Cluster 1**

（1）中等收入人士（平均收入等于 50000）；

（2）平均年龄52岁；

（3）受过教育（Graduation, 2n Cycle, Master, PhD）；

（4）没有家人的人，有孩子和没有孩子的家庭的人；

（5）经常买酒，也经常买肉；

（6）最常在网上购物；

（7）平均购买次数为 13。

**4.5.2 Cluster 2**

（1）高收入人士（平均收入等于 70000）；

（2）平均年龄为55岁；

（3）受过教育（Graduation, 2n Cycle, Master, PhD）；

（4）有孩子的家庭（Teenhome）；

（5）经常买酒，也经常买肉；

（6）大多数情况下自己在商店购买；

（7）最常购买（与其他相比）。

**4.5.3 Cluster 3**

（1）平均收入等于38000；

（2）平均年龄49岁；

（3）受过教育（Graduation, 2n Cycle, Master, PhD）；

（4）有孩子和没有孩子的家庭的人；

（5）购买次数少，因此在购买上花费的钱很少。

参考文献

[1] https://www.kaggle.com/datasets/imakash3011/customer-personality-analysis

[2] Pedro Domingos. A Few Useful Things to Know About Clustering[J].Communications of the ACM, vol. 52, no. 1, 2009, pp. 108-113.

[3] David Arthur and Sergei Vassilvitskii. K-means++: The Advantages of Careful Seeding[J]. Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, 2007, pp. 1027-1035.

[4] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise[J]. Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, 1996, pp. 226-231.

[5] Andrew Ng, Michael Jordan, and Yair Weiss. Spectral Clustering: Analysis and Algorithm[J]. Advances in Neural Information Processing Systems, vol. 14, 2002, pp. 849-856.

[6] Peter J. Rousseeuw. Hierarchical Clustering Algorithms[J]. Computer Science and Data Analysis Series, WIREs Computational Statistics, vol. 1, no. 1, 2009, pp. 82-90.

[7] Ward, J. H. (1963). Hierarchical grouping to optimize an objective function. Journal of the American statistical association, 58(301), 236-244.

[8] Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of computational and applied mathematics, 20, 53-65.

[9] Zhang, F., & Yang, J. (2020). A comprehensive review of spectral clustering algorithm: Techniques and applications. Expert Systems with Applications, 138, 112833.

[10] Kaufman, L., & Rousseeuw, P. J. (1990). Finding groups in data: an introduction to cluster analysis. John Wiley & Sons.

[11] <https://scikitlearn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html>

[12] <https://scikit-learn.org/stable/modules/generated/sklearn.cluster.SpectralClustering.html>

[13] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.silhouette\_score.html