### 10. 聚类分析 + 因子分析案例：城市宜居度分类

**问题背景**：某研究机构需对 20 个城市的宜居度分类，指标含 8 项（如人均 GDP、空气质量、教育资源、房价等），指标间存在相关性（如 “人均 GDP 高的城市房价也高”）。

**数据**：

* 20 个城市的 8 项指标数据（标准化后）：如人均 GDP（0-100 分）、空气质量指数（0-100 分，值越低越好）等。

**要求**：用因子分析提取 2-3 个公共因子（如 “经济因子”“环境因子”），基于因子得分用 K-means 聚类为 3 类，分析每类城市的宜居特征及改进方向。

### 10. 聚类分析 + 因子分析代码：城市宜居度分类

|  |
| --- |
| import numpy as np  import pandas as pd  import matplotlib.pyplot as plt  from sklearn.decomposition import FactorAnalysis  from sklearn.cluster import KMeans  from sklearn.preprocessing import StandardScaler  # 1. 数据准备（20个城市的8项指标）  np.random.seed(42)  n\_cities = 20  # 指标：人均GDP(万元)、空气质量指数、教育资源(学校数量)、医疗资源(医院数量)  # 房价(万元/㎡)、交通便利性(公交站点数)、绿化覆盖率(%)、就业率(%)  data = np.array([  [8.2, 65, 120, 80, 3.5, 500, 40, 92],  [6.8, 72, 95, 65, 2.8, 420, 35, 88],  [9.5, 58, 150, 95, 4.2, 600, 45, 94],  [5.2, 85, 70, 45, 1.9, 300, 25, 82],  [7.6, 68, 110, 75, 3.2, 480, 38, 90],  [4.8, 90, 65, 40, 1.7, 280, 22, 80],  [8.8, 60, 130, 85, 3.8, 550, 42, 93],  [5.5, 80, 75, 50, 2.1, 320, 28, 83],  [7.0, 70, 100, 70, 3.0, 450, 36, 89],  [6.2, 75, 85, 60, 2.5, 380, 32, 86],  [10.2, 55, 160, 100, 4.5, 650, 48, 95],  [4.5, 95, 60, 35, 1.5, 250, 20, 78],  [8.0, 66, 115, 82, 3.4, 490, 39, 91],  [5.8, 78, 80, 55, 2.3, 350, 30, 85],  [6.5, 73, 90, 62, 2.7, 400, 34, 87],  [9.2, 59, 140, 90, 4.0, 580, 44, 93],  [5.0, 88, 68, 42, 1.8, 290, 24, 81],  [7.3, 69, 105, 72, 3.1, 460, 37, 89],  [8.5, 63, 125, 88, 3.7, 530, 41, 92],  [6.0, 76, 88, 58, 2.4, 370, 33, 84]  ])  cities = [f'城市{i+1}' for i in range(n\_cities)]  indicators = ['人均GDP', '空气质量指数', '教育资源', '医疗资源',  '房价', '交通便利性', '绿化覆盖率', '就业率']  # 2. 数据标准化  scaler = StandardScaler()  data\_scaled = scaler.fit\_transform(data)  # 3. 因子分析（提取公共因子）  fa = FactorAnalysis(n\_components=3, random\_state=42)  factor\_scores = fa.fit\_transform(data\_scaled)  # 输出因子载荷（反映指标与因子的相关性）  loadings = pd.DataFrame(  fa.components\_.T,  index=indicators,  columns=[f'公共因子{i+1}' for i in range(3)]  )  print("因子载荷矩阵（指标与公共因子的相关性）：")  print(loadings.round(3))  # 4. 基于因子得分的聚类分析  kmeans = KMeans(n\_clusters=3, random\_state=42)  clusters = kmeans.fit\_predict(factor\_scores)  # 5. 结果分析  result = pd.DataFrame(data, columns=indicators)  result['城市'] = cities  result['聚类标签'] = clusters  # 输出各类别城市的特征  print("\n各类别城市的平均指标：")  for cluster in range(3):  cluster\_data = result[result['聚类标签'] == cluster]  print(f"\n类别{cluster+1}（城市数量：{len(cluster\_data)}）：")  print(cluster\_data.iloc[:, :-2].mean().round(2))  # 6. 可视化聚类结果（前2个因子）  plt.figure(figsize=(10, 6))  plt.scatter(factor\_scores[:, 0], factor\_scores[:, 1], c=clusters, cmap='viridis', s=100)  plt.scatter(kmeans.cluster\_centers\_[:, 0], kmeans.cluster\_centers\_[:, 1],  marker='X', s=300, c='red', label='聚类中心')  plt.xlabel('公共因子1（经济与资源）')  plt.ylabel('公共因子2（环境与就业）')  plt.title('城市宜居度聚类结果')  plt.legend()  plt.grid(True)  plt.show() |