计算机科学技术学院实验报告

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **课程名称** | 机器学习 | | | **学 号** | 230511637 |
| **实验项目** | 对航空公司客户价值进行聚类分析 | | | **姓 名** | 张世浩 |
| **学 时** | 1.5h | **项目性质** | 设计型 | **班 级** | 2305116 |
| **指导教师** | 王玲,高宁 | **实验地点** | 实训楼442 | **日 期** | 2025年10月26日 |
| 1. **实验目的和要求**   会用Python创建Kmeans聚类分析模型  使用KMeans模型对航空公司客户价值进行聚类 对预测结果进行评价  会对聚类结果进行分析评价 | | | | | |
| 1. **实验环境**   Anconda3  Python3.11 | | | | | |
| 1. **实验内容与过程**   **实验内容:**  1. 使用sklearn.cluester的KMeans类对航空公司客户数据进行聚类分析， 把乘客分到不同的类别中 ⎫ 数据集：air\_data.csv ⎫ 数据集大小：62052条不重复数据 ⎫ 原数据有40个属性，为了大家训练模型方便，本实验使用预处理后的标准化数据， 该数据有5个属性  2. 数据说明 ：  ZL：入会至当前时长，反应可能的活跃时间 λ  ZR：最近消费时间间隔，反应最近一段时间活跃程度 λ  ZF：消费频次，反应客户忠诚度 λ ZM：消费里程总额，反应客户对乘机的依赖程度 λ  ZC：舱位等级对应折扣系数，一般舱位等级越高，折扣系数越大  载入训练数据、显示读入数据的前5行 1  使用肘部法确定最佳聚类的类别数目,输出最佳类别数目和对应的折线图  3. 训练KMeans聚类模型，把数据聚成最佳类别数  训练KMeans聚类模型，把数据聚成最佳类别数  画出聚类中心点在每个维度上的散点图，并按统一类别把聚类中心用线连接起来分析聚类结果  **流程图:**    代码:  import pandas as pd import numpy as np import matplotlib.pyplot as plt from sklearn.cluster import KMeans  # 解决中文显示问题 plt.rcParams['font.sans-serif'] = ['SimHei', 'Microsoft YaHei'] plt.rcParams['axes.unicode\_minus'] = False  # 1. 载入训练数据 data = pd.read\_csv('air\_data.csv') X = data[['ZL', 'ZR', 'ZF', 'ZM', 'ZC']] print("测试数据前5条:") print(X.head()) print("="\*50)  # 2. 肘部法确定最佳聚类数 wcss = [] for i in range(1, 11):  kmeans = KMeans(  n\_clusters=i,  init='k-means++',  max\_iter=300,  n\_init=10,  random\_state=0,  algorithm='lloyd'  )  kmeans.fit(X)  wcss.append(kmeans.inertia\_)  # 肘部法折线图（含学号姓名） plt.figure(figsize=(10, 6)) plt.plot(range(1, 11), wcss, marker='o', linestyle='-', color='b') plt.title('肘部法确定最佳聚类数\n学号：230511637 姓名：张世浩') # 中文标题更直观 plt.xlabel('聚类类别数目') plt.ylabel('组内平方和 (WCSS)') plt.grid(True, alpha=0.3) plt.show()  bestClass = 5 print(f"最佳聚类类别数目: {bestClass}") print("="\*50)  # 3. 训练KMeans模型 kmeans\_model = KMeans(  copy\_x=True,  init='k-means++',  max\_iter=300,  n\_clusters=bestClass,  n\_init=10,  random\_state=None,  tol=0.0001,  verbose=0,  algorithm='lloyd' ) cluster\_labels = kmeans\_model.fit\_predict(X) data['Cluster\_Label'] = cluster\_labels  # 4. 输出聚类结果 cluster\_sample\_count = data['Cluster\_Label'].value\_counts().sort\_index() print("每个聚类类别的样本数量:") print(cluster\_sample\_count) print("="\*50)  cluster\_centers = pd.DataFrame(  kmeans\_model.cluster\_centers\_,  columns=['ZL', 'ZR', 'ZF', 'ZM', 'ZC'],  index=[f'Cluster\_{i}' for i in range(bestClass)] ) print("每个聚类类别的中心点:") print(cluster\_centers) print("="\*50)  # 聚类中心点散点图（含学号姓名） plt.figure(figsize=(12, 7)) feature\_positions = np.arange(len(cluster\_centers.columns)) colors = ['r', 'g', 'b', 'y', 'purple']  for i in range(bestClass):  cluster\_center = cluster\_centers.iloc[i].values  plt.scatter(  feature\_positions,  cluster\_center,  color=colors[i],  label=f'Cluster\_{i}',  s=100  )  plt.plot(  feature\_positions,  cluster\_center,  color=colors[i],  linestyle='-',  linewidth=2  )  plt.xlabel('特征维度 (ZL:入会时长, ZR:最近消费间隔, ZF:消费频次, ZM:消费里程, ZC:舱位折扣系数)') plt.ylabel('标准化中心值') plt.title('各特征维度的聚类中心分布\n学号：230511637 姓名：张世浩') # 中文标题更直观 plt.xticks(feature\_positions, cluster\_centers.columns) plt.legend() plt.grid(True, alpha=0.3) plt.show()  # 5. 聚类结果分析 print("聚类结果分析（基于实验定义的5个核心属性含义）:") for i in range(bestClass):  center = cluster\_centers.iloc[i]  print(f"\nCluster\_{i}:")  print(f"- 入会时长(ZL): {center['ZL']:.4f}（值越高→入会时间越长，潜在活跃时间越久）")  print(f"- 最近消费间隔(ZR): {center['ZR']:.4f}（值越低→最近消费越近，近期活跃度越高）")  print(f"- 消费频次(ZF): {center['ZF']:.4f}（值越高→消费次数越多，客户忠诚度越高）")  print(f"- 消费里程(ZM): {center['ZM']:.4f}（值越高→总里程越多，对乘机依赖度越高）")  print(f"- 舱位折扣系数(ZC): {center['ZC']:.4f}（值越高→舱位等级越高，客户消费能力越强）") | | | | | |
| 1. **实验结果与分析**   **实验结果截图:**      **实验结果分析:**  **一、实验准备阶段**  **1. 环境配置**  本次实验基于 Python 环境开展，需提前安装以下库：   * 数据处理库：pandas（用于数据读取与处理）、numpy（用于数值计算） * 可视化库：matplotlib（用于绘制实验图表） * 机器学习库：scikit-learn（提供 KMeans 聚类算法）   **2. 数据准备**  实验数据为航空客户特征数据集 air\_data.csv，包含 5 个核心特征：   * ZL：入会时长（客户注册会员至当前的时间长度） * ZR：最近消费间隔（客户最后一次消费至当前的时间间隔） * ZF：消费频次（客户累计消费次数） * ZM：消费里程（客户累计飞行里程） * ZC：舱位折扣系数（客户购买舱位的平均折扣水平，值越高代表舱位等级越高）   **二、实验实施步骤**  **步骤 1：数据加载与初步观察**   1. 通过 pandas 库读取 air\_data.csv 数据：   data = pd.read\_csv ('air\_data.csv')   1. 提取 5 个核心特征作为聚类输入数据：   X = data [['ZL', 'ZR', 'ZF', 'ZM', 'ZC']]   1. 查看数据前 5 条记录，确认数据格式与特征完整性：   print ("测试数据前 5 条:")  print (X.head ())  **步骤 2：确定最佳聚类数（肘部法）**   1. 设定聚类数范围为 1-10，循环训练 KMeans 模型并计算组内平方和（WCSS）：wcss = [] # 存储不同聚类数对应的组内平方和for i in range (1, 11):kmeans = KMeans (n\_clusters=i,init='k-means++', # 优化初始质心选择max\_iter=300, # 最大迭代次数n\_init=10, # 多次初始化取最优结果random\_state=0, # 固定随机种子，保证结果可复现algorithm='lloyd' # 经典 KMeans 算法)kmeans.fit (X)wcss.append (kmeans.inertia\_) # 记录当前聚类数的 WCSS 2. 绘制肘部法折线图，分析最佳聚类数：plt.figure (figsize=(10, 6))plt.plot (range (1, 11), wcss, marker='o', linestyle='-', color='b')plt.title (' 肘部法确定最佳聚类数 \n 学号：230511637 姓名：张世浩 ')plt.xlabel (' 聚类类别数目 ')plt.ylabel (' 组内平方和 (WCSS)')plt.grid (True, alpha=0.3)plt.show () 3. 根据折线图 "拐点" 判断，确定最佳聚类数为 5 类：bestClass = 5   **步骤 3：训练 KMeans 聚类模型**   1. 使用最佳聚类数（5 类）构建 KMeans 模型并训练：kmeans\_model = KMeans (copy\_x=True,init='k-means++',max\_iter=300,n\_clusters=bestClass, # 设定最佳聚类数n\_init=10,random\_state=None,tol=0.0001,verbose=0,algorithm='lloyd') 2. 执行聚类并获取每个样本的聚类标签：cluster\_labels = kmeans\_model.fit\_predict (X) 3. 将聚类标签添加到原始数据中，便于后续分析：data ['Cluster\_Label'] = cluster\_labels   **步骤 4：聚类结果可视化与统计**   1. 统计每个聚类类别的样本数量：cluster\_sample\_count = data ['Cluster\_Label'].value\_counts ().sort\_index ()print ("每个聚类类别的样本数量:")print (cluster\_sample\_count) 2. 提取并展示各聚类的中心点（特征均值）：cluster\_centers = pd.DataFrame (kmeans\_model.cluster\_centers\_,columns=['ZL', 'ZR', 'ZF', 'ZM', 'ZC'],index=[f'Cluster\_{i}' for i in range (bestClass)])print ("每个聚类类别的中心点:")print (cluster\_centers) 3. 绘制聚类中心点特征分布折线图，直观对比不同聚类的特征差异：plt.figure (figsize=(12, 7))feature\_positions = np.arange (len (cluster\_centers.columns))colors = ['r', 'g', 'b', 'y', 'purple'] # 为每个聚类分配独特颜色   for i in range(bestClass):cluster\_center = cluster\_centers.iloc[i].values  **绘制散点标记中心点**  plt.scatter(feature\_positions,cluster\_center,color=colors[i],label=f'Cluster\_{i}',s=100)  **绘制折线连接同一聚类的特征值**  plt.plot(feature\_positions,cluster\_center,color=colors[i],linestyle='-',linewidth=2)  plt.xlabel (' 特征维度 (ZL: 入会时长，ZR: 最近消费间隔，ZF: 消费频次，ZM: 消费里程，ZC: 舱位折扣系数)')plt.ylabel (' 标准化中心值 ')plt.title (' 各特征维度的聚类中心分布 \n 学号：230511637 姓名：张世浩 ')plt.xticks (feature\_positions, cluster\_centers.columns)plt.legend ()plt.grid (True, alpha=0.3)plt.show ()  **步骤 5：聚类结果分析**  基于聚类中心点的特征数值，结合各特征的业务含义（如 "ZR 值越低代表近期活跃度越高"），逐类分析客户群体特征：print ("聚类结果分析（基于实验定义的 5 个核心属性含义）:")for i in range (bestClass):center = cluster\_centers.iloc [i]print (f"\nCluster\_{i}:")print (f"- 入会时长 (ZL): {center ['ZL']:.4f}（值越高→入会时间越长，潜在活跃时间越久）")print (f"- 最近消费间隔 (ZR): {center ['ZR']:.4f}（值越低→最近消费越近，近期活跃度越高）")print (f"- 消费频次 (ZF): {center ['ZF']:.4f}（值越高→消费次数越多，客户忠诚度越高）")print (f"- 消费里程 (ZM): {center ['ZM']:.4f}（值越高→总里程越多，对乘机依赖度越高）")print (f"- 舱位折扣系数 (ZC): {center ['ZC']:.4f}（值越高→舱位等级越高，客户消费能力越强）")  **三、实验注意事项**   1. 中文显示设置：实验前需通过 plt.rcParams 配置中文字体（如 SimHei、Microsoft YaHei），避免图表中中文乱码。 2. 模型参数稳定性：KMeans 算法对初始质心敏感，通过 n\_init=10 设置多次初始化，取最优结果以保证稳定性。 3. 结果可复现性：在确定最佳聚类数阶段固定 random\_state=0，确保肘部法结果可复现；最终训练阶段可放开随机种子，允许模型自适应优化 | | | | | |
| 1. **实验心得**   本次对航空公司客户价值进行聚类分析的实验，让我深入理解了 KMeans 聚类算法的实际应用流程，每一步操作的设计都有其明确的目的和逻辑，具体心得如下：  **1. 数据加载与预处理：明确分析对象，为模型奠基**  **操作**：通过pandas读取air\_data.csv数据，选取ZL（入会时长）、ZR（最近消费间隔）、ZF（消费频次）、ZM（消费里程）、ZC（舱位折扣系数）5 个预处理后的标准化特征作为分析对象，并打印前 5 行数据。**原因**：   * 原始数据包含 40 个属性，直接分析会增加模型复杂度且可能引入冗余信息。选取的 5 个特征经过预处理（标准化），已去除量纲影响，能更公平地反映各维度对客户价值的贡献。 * 打印前 5 行数据是为了快速验证数据读取是否正确，确认特征名称和格式是否与预期一致，避免后续分析因数据错误导致结果偏差。   **2. 肘部法确定最佳聚类数：科学选择 K 值，平衡聚类效果与复杂度**  **操作**：通过循环训练不同k值（1 到 10）的 KMeans 模型，计算并存储每个k对应的组内平方和（WCSS），绘制折线图，根据 “肘部” 位置确定最佳聚类数k=5。**原因**：   * KMeans 算法需要预先指定聚类数k，但k的选择直接影响聚类结果的合理性。WCSS（组内平方和）反映了同类样本的紧凑程度，k增大时 WCSS 会逐渐减小（样本被划分得更细），但当k超过合理值后，WCSS 下降幅度会明显放缓，形成 “肘部”，此时的k是平衡聚类效果（紧凑性）和模型复杂度（避免过拟合）的最优解。 * 绘制折线图能直观展示 WCSS 随k的变化趋势，便于快速定位 “肘部” 位置，相比仅凭经验选择k更具科学性。   **3. 训练 KMeans 模型：执行聚类，实现客户分群**  **操作**：使用确定的最佳k=5初始化 KMeans 模型，通过fit\_predict方法对数据进行聚类，得到每个样本的聚类标签，并将标签添加到原始数据中。**原因**：   * KMeans 算法通过 “初始化聚类中心→分配样本→更新中心→迭代收敛” 的流程，将相似样本聚为一类。指定n\_init=10和init='k-means++'是为了避免初始中心随机选择导致的局部最优问题，提高模型稳定性。 * 为样本添加聚类标签后，可方便后续统计每个类别的样本数量，分析各类别的规模分布，为客户分群的商业价值评估提供基础。   **4. 聚类结果可视化与统计：直观呈现分群特征**  **操作**：统计每个聚类的样本数量，计算并打印聚类中心，绘制聚类中心在 5 个特征维度上的散点图（并用线连接同类中心）。**原因**：   * 样本数量统计能反映各类别的客户规模，规模较大的类别可能是企业的核心客户群体或待优化群体，具有更高的分析优先级。 * 聚类中心是每个类别的 “代表性样本”，其特征值反映了该类客户的整体特征。通过散点图和连线可视化，可直观对比不同类别在 5 个维度上的差异（如哪些类别消费频次高、哪些最近消费间隔短），为后续客户价值分析提供直观依据。   **5. 聚类结果分析：挖掘客户价值，指导商业决策**  **操作**：结合 5 个特征的业务含义（如ZR越低表示近期越活跃，ZC越高表示消费能力越强），逐一分析每个聚类的特征，总结各类客户的价值属性。**原因**：   * 聚类的最终目的是为业务决策提供支持。例如：若某类客户ZF（消费频次）和ZM（消费里程）高、ZR（最近消费间隔）低，说明是高价值活跃客户，企业应通过会员福利维持其忠诚度；若某类客户ZL（入会时长）长但ZR高、ZF低，可能是流失风险客户，需通过营销活动唤醒。 * 只有结合业务场景解读聚类结果，才能将技术分析转化为可执行的商业策略，体现机器学习在实际业务中的价值。   通过本次实验，我不仅掌握了 KMeans 聚类的代码实现，更理解了 “数据加载→参数选择→模型训练→结果可视化→业务解读” 全流程的逻辑闭环。每一步操作都服务于 “准确分群并挖掘客户价值” 的核心目标，而科学的方法（如肘部法）和可视化工具则是实现这一目标的关键手段。未来在类似分析中，我会更注重结合业务背景设计特征和解读结果，让机器学习模型真正为决策赋能。 | | | | | |
| 1. **教师评语** | | | | | |
| 1. **实验成绩**   教师签名：王玲,高宁 批阅日期： 2025年 10月 26日 | | | | | |

注：项目性质为 演示型、验证型、设计型、综合型和创新型。