百货商场数据分析

互联网的发展，实体商业受到了巨大的冲击。网上购物成了很多人的首选，这使得对客流量十分依赖的百货商场主动或被动地研究新的营销方案。在这个互联网时代，要与电商竞争的百货商场必须要向电商学习，应用大数据技术于营销活动之中。

营销的核心是人，百货商场的营销对象是它的顾客，尤其是那些愿意或已经成为其会员的顾客。若想营销活动高效执行且把成本控制到可接受的范围，那么主动去了解顾客对百货商场来说就显得极为重要。

百货商场的顾客很多，传统的发放调查问卷研究方法难以覆盖总体，且费时费力，效率低下。而采用大数据技术，直接对顾客的订单记录进行分析，再对顾客中的会员进行用户画像，可以直接覆盖总体，还可以有不同消费特点的会员划分，向其定制相应的营销方案，不仅使营销方案更高效，同时也节约了营销成本。

**学习目标**

1. 百货商场数据理解以及建模流程
2. 数据探索与预处理
3. 统计分析
4. 会员用户画像
5. 会员用户细分及营销方案制定

目录

[百货商场数据分析 1](#_Toc47822214)

[1 百货商场数据理解与建模流程 4](#_Toc47822215)

[1.1 了解百货商场数据 4](#_Toc47822216)

[1.2 实现百货商场数据分析的步骤与方法 5](#_Toc47822217)

[2 数据探索与预处理 5](#_Toc47822218)

[2.1 利用Python进行数据预处理的方法 6](#_Toc47822219)

[2.2 本案例用到的数据预处理方法 6](#_Toc47822220)

[2.3 任务实现 7](#_Toc47822221)

[2.3.1 对“cumcm2018c1.xlsx”预处理源码 7](#_Toc47822222)

[2.3.2 对“cumcm2018c2.csv”预处理源码 8](#_Toc47822223)

[2.3.3 合并数据源代码 9](#_Toc47822224)

[3 统计分析 10](#_Toc47822225)

[3.1 按年龄构成区分会员 10](#_Toc47822226)

[3.2 统计不同年龄段的消费金额 11](#_Toc47822227)

[3.3 统计不同性别的会员的人数和消费金额 12](#_Toc47822228)

[3.4 统计会员与非会员的订单数量 12](#_Toc47822229)

[3.5 统计会员与非会员的消费金额 13](#_Toc47822230)

[3.6 统计不同年份的会员消费额 15](#_Toc47822231)

[3.7 统计不同年份的每月会员消费额 16](#_Toc47822232)

[3.8 统计一日中不同时间段的消费额 18](#_Toc47822233)

[3.9 统计每年各个季度的会员消费人次 19](#_Toc47822234)

[3.10 任务实现 21](#_Toc47822235)

[3.10.1 按年龄段区分会员源代码 21](#_Toc47822236)

[3.10.2 按统计不同年龄段的消费金额源代码 21](#_Toc47822237)

[3.10.3 统计不同性别的会员的人数和消费金额源代码 22](#_Toc47822238)

[3.10.4 统计会员与非会员订单数量源代码 24](#_Toc47822239)

[3.10.5 统计会员与非会员消费金额源代码 24](#_Toc47822240)

[3.10.6 统计不同年份的会员消费额源代码 26](#_Toc47822241)

[3.10.7 统计不同年份会员每月消费额源代码 27](#_Toc47822242)

[3.10.8 统计一日中不同时间段的消费额源代码 28](#_Toc47822243)

[3.10.9 统计每年各个季度的会员消费人次源代码 29](#_Toc47822244)

[4 会员用户画像 30](#_Toc47822245)

[4.1 构建会员基本特征 31](#_Toc47822246)

[4.2 构建会员业务特征 31](#_Toc47822247)

[4.3 构建会员兴趣特征 32](#_Toc47822248)

[4.4 会员特征可视化 33](#_Toc47822249)

[4.5 任务实现 34](#_Toc47822250)

[4.5.1 构建会员基本特征源代码 34](#_Toc47822251)

[4.5.2 构建会员业务特征源代码 35](#_Toc47822252)

[4.5.3 构建会员兴趣特征源代码 36](#_Toc47822253)

[4.5.4 会员特征可视化源代码 37](#_Toc47822254)

[5 会员用户细分及营销方案制定 38](#_Toc47822255)

[5.1 构建会员价值分析特征指标 38](#_Toc47822256)

[5.2 K-Means聚类 39](#_Toc47822257)

[5.3 营销方案制定 41](#_Toc47822258)

[5.4 任务实现 42](#_Toc47822259)

[5.4.1 构建会员价值分析特征指标源代码 42](#_Toc47822260)

[5.4.2 k-means聚类建模源代码 43](#_Toc47822261)

[6 小结 44](#_Toc47822262)

# 百货商场数据理解与建模流程

**任务描述**

本次建模针对百货商场的会员信息表以及顾客订单记录表，在对数据进行去重、去缺失值、补缺失值、去除异常值、异常值修正等操作后，通过会员的基本特征、消费偏好特征、消费行为特征进行用户画像，并对会员的特征数据进行聚类，通过不同类的特点定制相应的营销方案。

**任务分析**

可以通过以下2个步骤了解百货商场购物数据与建模流程。

1. 了解百货商场购物数据和会员信息数据。
2. 实现百货商场数据分析的步骤与方法。

## 了解百货商场数据

本案例主要针对百货商场的购物情况表“cumcm2018c2.csv”，以及会员信息表“cumcm2018c1.xlsx”。

表1是购物情况表的部分数据，相关字段信息如下：

kh：会员卡号；dtime：消费时间；spbm：商品编码；sl：销售数量；sj：商品售价；je：消费金额；spmc：商品名称；jf：积分；syjh：收银机号；djh：单据号；gzbm：柜组编码；gzmc：柜组名称

表1

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| kh | dtime | spbm | sl | sj | je | spmc | jf | syjh | djh | gzbm | gzmc |
| 1be1e3fe | 2015-01-01 00:05:41.593 | f09c9303 | 1 | 290.0 | 270.2 | 兰芝化妆品正价瓶 | 270.2 | 6 | 25bb | 8077.0 | 兰芝柜 |
| 1be1e3fe | 2015-01-01 00:05:41.593 | f09c9303 | 1 | 325.0 | 302.8 | 兰芝化妆品正价瓶 | 302.8 | 6 | 25bb | 8077.0 | 兰芝柜 |

表2是会员信息表，相关字段信息如下：

kh：会员卡号；csrq：出生日期；xb：性别；rhsj：入会时间

|  |  |  |  |
| --- | --- | --- | --- |
| kh | csrq | xb | rhsj |
| 8fd77c69 | 1978/10/14 0:00 | 0 | 2002/2/25 0:00 |
| 393285c3 | 1968/5/9 0:00 | 0 | 2002/10/23 21:10 |
| ba52e695 | 1977/5/18 0:00 | 0 | 2002/10/24 11:06 |
| c497071b | 1970/8/14 0:00 | 0 | 2002/10/24 11:27 |
| ee68a6de | 1974/9/20 0:00 | 0 | 2002/10/24 11:31 |

## 实现百货商场数据分析的步骤与方法

主要步骤如下。

1. 数据探索与预处理
2. 数据统计
3. 数据可视化
4. 特征工程与模型建立
5. 制定营销方案

# 数据探索与预处理

**任务描述**

本案例需要使用python对“cumcm2018c1.xlsx”和“cumcm2018c2.csv”文件进行分析，包括数据分布、特征选择、缺失值处理等。涉及到的知识主要有pandas库、numpy库。

**任务分析**

对百货商场购物数据进行探索和预处理可以分为以下3个步骤。

1. 缺失值、异常值处理
2. 数据分布探索
3. 数据合并

## 利用Python进行数据预处理的方法

数据的预处理方法主要有数据清理、数据集成、数据转换、数据归约。

**数据清理**：数据清理例程通过填写缺失的值、光滑噪声数据、识别或删除离群点并解决不一致性来“清理”数据。主要是达到如下目标：格式标准化，异常数据清除，错误纠正，重复数据的清除。

**数据集成**：数据集成例程将多个数据源中的数据结合起来并统一存储，建立数据仓库的过程实际上就是数据集成。

**数据转换**：通过平滑聚集，数据概化，规范化等方式将数据转换成适用于数据挖掘的形式。

**数据归约**：数据挖掘时往往数据量非常大，在少量数据上进行挖掘分析需要很长的时间，数据归约技术可以用来得到数据集的归约表示，它小得多，但仍然接近于保持原数据的完整性，并结果与归约前结果相同或几乎相同。

## 本案例用到的数据预处理方法

对于会员记录表“cumcm2018c1.xlsx”，用到的预处理有去重复值、删除缺失值、填充缺失值、特殊值替代缺失值、特殊值替代异常值、特征重命名。

“cumcm2018c1.xlsx”中拼音简写的字段名被替换成易于读懂的中文字段名，会员卡号重复的保留第一条记录其余删除，出生日期缺失的、在1900年之前出生的都用特殊值“1800/01/01”替代，出生日期超过2018年的作为异常值删除，性别信息缺失的由于记录较少所以删除，入会时间缺失的用众数填补。并将预处理结果保留为“会员信息表.csv”，出生日期正常的记录提取出来并保存到“有出生日期的会员信息表.csv”。

对于消费记录表“cumcm2018c2.csv”，用到的预处理方法有特征重命名、筛选特征、数据区分。

“cumcm2018c2.csv”中的拼音简写字段名被替换成易懂的中文字段名，销售记录重复的予以删除，消费时间异常的予以删除，与分析目标无关的字段（柜组名称、柜组编码、收银机号）予以删除，其中会员卡号缺失的例句是非会员的消费记录，将会员与非会员的消费记录分开，会员消费记录中有缺失值的予以删除。并将会员消费记录保存在“会员购物情况表.csv”，非会员的消费记录保存在“会员购物情况表.csv”。

次外，将“会员购物情况表.csv”与“会员信息表.csv”按会员卡号合并，合并方式为外连接，并将合并后的数据保存在“task1.csv”中。以上的操作依赖python的pandas库。

## 任务实现

### 对“cumcm2018c1.xlsx”预处理源码

代码 2‑1 对“cumcm2018c1.xlsx”预处理

|  |
| --- |
| '''  对cumcm2018c2.csv文件进行预处理，预处理包括：  ①更改中文列名  ②去除重复值  ③去除异常值  ④筛选特征  ⑤将会员数据和非会员暑假区分  ⑥会员暑假生成“会员购物情况表.csv”，非会员数据生成“非会员购物情况表.csv”  '''  import pandas as pd  #读取数据  data = pd.read\_csv('../../data/cumcm2018c2.csv', low\_memory=False)  #更改列名  columns = ['会员卡号','消费时间','商品编码','销售数量','商品售价',  '消费金额','商品名称','积分','收银机号',  '单据号','柜组编码','柜组名称']  data.columns = columns  #去除重复值  before\_shape = data.shape  print('去重复值之前的数据：',before\_shape)  data.drop\_duplicates(inplace=True)  after\_shape = data.shape  print('去缺失值后的数据：',after\_shape)  print('共去除了{}条数据'.format(before\_shape[0]-after\_shape[0]))  del before\_shape,after\_shape  #去除异常值  data['消费时间'] = pd.to\_datetime(data['消费时间'])  #print(data.loc[data['消费时间'].dt.year==2019].shape)  #经查，数据皆是2015、2016、2017、2018年的，并无2019年数据  #筛选特征  data.drop(['柜组名称','柜组编码','收银机号'],axis=1,inplace=True)  #缺失值处理  print(data.isnull().sum())  print('去除缺失值前的数据：',data.shape)  non\_vip\_data = data[data.isnull().values==True].copy()  data.dropna(axis=0,how='any',inplace=True)  print('去缺失值后的数据：',data.shape)  #保存预处理结果  data.to\_csv('..\\..\\data\\会员购物情况表.csv',index=False)  non\_vip\_data.to\_csv('..\\..\\data\\非会员购物情况表.csv',index=False) |

**\*代码详见：D:\python\_files\百货商场数据分析\项目任务\code\task\_1\task1-1\_1.py\***

### 对“cumcm2018c2.csv”预处理源码

代码 2‑2 对“cumcm2018.c2.csv”预处理

|  |
| --- |
| '''  对“cumcm2018c1.xlsx”进行预处理，预处理包括：  ①去除卡号重复的记录  ②将出生日期异常的通通标为1800年1月1日，以备用  ③入会时间缺失的用众数填补  ④将数据生成“会员信息表.csv”  ⑤把出生日期正常的记录单独生成“有出生日期的会员信息表.csv”，以备用  '''  import pandas as pd  # 1.读取数据  data2 = pd.read\_excel('..\\..\\data\\cumcm2018c1.xlsx')  print('预处理之前的记录数量：',data2.shape[0])  # 2.去除重复的会员卡号  before\_shape = data2.shape  print('去重复值之前的数据：',before\_shape)  data2.drop\_duplicates('会员卡号',inplace=True)  after\_shape = data2.shape  print('去缺失值后的数据：',after\_shape)  print('共去除了{}条数据'.format(before\_shape[0]-after\_shape[0]))  del before\_shape,after\_shape  # 3.检查并去除异常值  ## 3.1出生日期用特殊值1800/01/01填补缺失值  data3 = data2[~data2['出生日期'].isnull().values==True].copy() #出生日期为非空值的会员数据  print('有出生日期记录的会员数量有：{}条'.format(data3.shape[0]))  print('没有出生日期记录的会员数量有：{}条'.format(data2.shape[0]-data3.shape[0]))  data2['出生日期'] = data2['出生日期'].fillna('1800/01/01')  data2['出生日期'] = pd.to\_datetime(data2['出生日期'])  ## 3.2用特殊出生日期1800/01/01替代异常的出生日期  print('1900年之前出生的定义为异常值，异常值有{}条'\  .format(data2[data2['出生日期'].dt.year<1900].shape[0]))  data2.loc[(data2['出生日期'].dt.year>1800) & \  (data2['出生日期'].dt.year<1900),'出生日期'] = pd.to\_datetime('1800/01/01')  ## 3.3 用众数填补入会时间缺失值  mode\_year = int(data2['入会时间'].dt.year.mode())  data2['入会时间'] = data2['入会时间'].fillna('{}/01/01'.format(mode\_year))  data2['入会时间'] = pd.to\_datetime(data2['入会时间'])  ## 3.4手动删除入会时间为1900年和2080年的3条数据  ## 3.5删除性别不详的会员记录  print('性别不详的会员数量：',data2[data2['性别'].isnull().values==True].shape[0])  data2 = data2[~data2['性别'].isnull().values==True]  # 4.保存预处理结果  print('预处理完之后剩下的记录数量：',data2.shape[0])  data2.to\_csv('..\\..\\data\\会员信息表.csv',index=False)  print('有出生日期记录的会员数量：',data3.shape[0])  data3.to\_csv('..\\..\\data\\有出生日期的会员信息表.csv',index=False) |

**\*代码详见：D:\python\_files\百货商场数据分析\项目任务\code\task\_1\task1-1\_2.py\***

### 合并数据源代码

代码 2‑3 将“会员购物情况表”与“会员信息表合并”

|  |
| --- |
| '''  把“会员信息表.csv”和“会员购物情况表.csv”合并成“task1.csv”，合并方式为外连接  '''  import pandas as pd  table\_1 = pd.read\_csv('../../data/会员信息表.csv')  table\_2 = pd.read\_csv('../../data/会员购物情况表.csv')  table = pd.merge(table\_1,table\_2,how='outer',on='会员卡号')  table.to\_csv('..\\..\\data\\task1.csv',index=False) |

**\*代码详见：D:\python\_files\百货商场数据分析\项目任务\code\task\_1\task1-2\_1.py\***

# 统计分析

**任务描述**

统计购物情况表中的数据，通过分析男女会员之比、男女消费金额之比、会员年龄构成、消费时间段分布、消费季度分布等数据，用python的matplotlib库进行相关图形的可视化，直观展现会员购物行为特征的分布，以便更好地了解数据。

**任务分析**

对购物情况数据进行统计分析可分为以下9个步骤

1. 将会员按年龄分为老年人、中年人、青年人，并统计年龄构成
2. 针对会员用户，统计不同年龄段的消费金额
3. 统计不同性别的消费金额比例和人数之比
4. 统计会员与非会员订单数量
5. 统计会员与非会员消费金额
6. 统计不同年份的会员消费金额
7. 按不同年份统计会员每月消费金额
8. 将一天分为凌晨、上午、中午、下午、晚上，并统计各时间段的会员消费金额
9. 根据消费时间，按年份统计各个季节的消费人数

## 按年龄构成区分会员

中国发展迅速，不同年龄段的人消费观有很大差异，消费观的差异直接体现在消费行为中。会员信息表中有会员的出生日期，消费日期减去会员的出生日期即可得到会员的真实年龄。按照经验方法，把45岁以下的定义为青年人，45岁到60岁的定义为中年人，60岁以上的定义为老年人。

在预处理过程中，出生日期完整的会员信息已经保存到“有出生日期的会员信息表.csv”，接下来用2018减去他们的出生年份便可得到真实年龄，得到年龄后绘制饼状图，便可得到会员的年龄构成，如图1所示。

由图1可知，会员中人数占比最高的是青年人，其次是老年人，占比最低的是中年人。

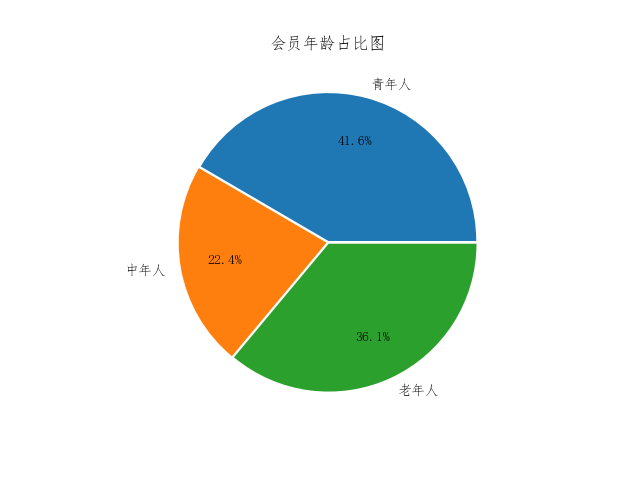


图3-1

## 统计不同年龄段的消费金额

不同年龄段的会员人数并不能直接体现消费行为特征，相比于中年人和老年人，青年人更愿意接受新事物，因此青年人会员数量会比较多。下面统计不同年龄段的会员的消费金额，以分析消费特征。

预处理中已经把会员信息表和购物情况表合并为“task1.csv”，“task1.csv”中包含消费行为的发生时间和会员的出生日期，二者相减得到消费发生时的真实年龄，再按照年龄将会员分为青年人、中年人、老年人，并将三者的消费金额加总，得到这三个年龄段的会员在2015年到2018年的消费总金额数据，将数据可视化得到图3-2-1和图3-2-1。

由图3-2-1可知，总消费金额的一半以上是由数量不到一半的青年人会员贡献的，而数量最少的只有22.4%的中年人会员也贡献了46%的消费金额，数量占比36.1%的老年人会员贡献的消费金额只有3.4%。消费主力是青年人和中年人。

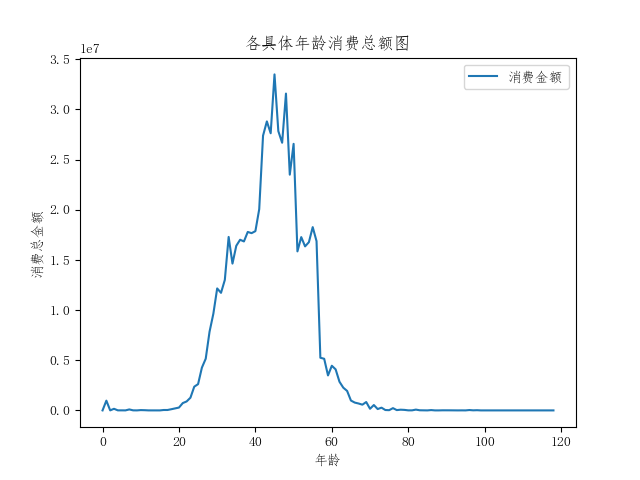
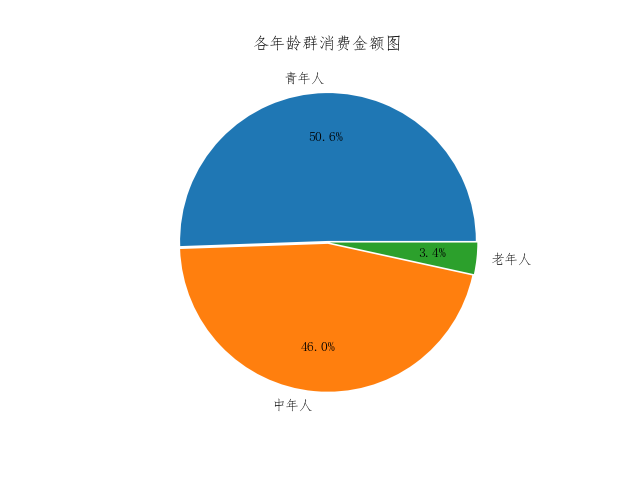


图3-2-1 图3-2-2

## 统计不同性别的会员的人数和消费金额

不同性别的会员消费行为可能会有差异，以下通过统计不同性别的会员人数以及其消费金额，分析消费行为在性别上的差异。

“会员信息表.csv”中有性别字段，使用pandas的groupby函数便可得到不同性别的会员的人数，接着可用matplotlib库进行可视化，得到图3-3-1。“task1.csv”中有性别字段和消费金额字段，同样使用pandas的groupby函数可得到不同性别的消费金额，再用matplotlib进行可视化得到图3-3-2。

由以下两图可看出，无论是数量还是消费金额，女会员都远超男会员。

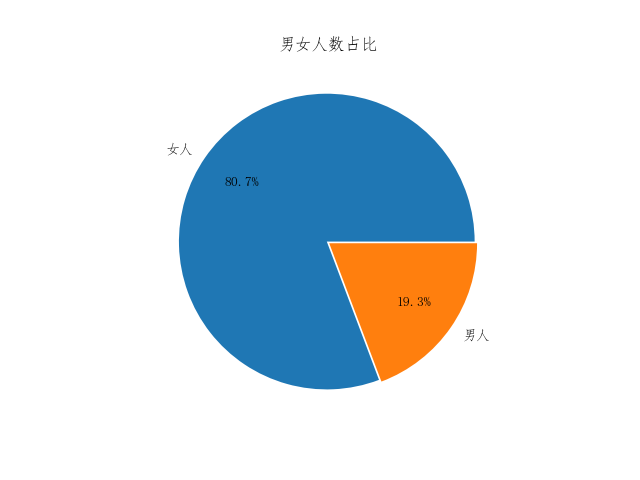


图3-3-1 图3-3-2

## 统计会员与非会员的订单数量

一般来说，会员顾客比非会员顾客的黏性更高，即成为回头客的几率更大，我们甚至可以把会员定义为熟客，熟客越多往往代表着经营状况越好。以下统计2015到2018年的订单中，会员订单与非会员订单的数量之比。

在数据预处理阶段，本文按照字段“会员卡号”是否为缺失值，区分了会员消费记录和非会员消费记录，并分别保存在“会员购物情况表.csv”和“非会员购物情况表.csv”中。使用pandas读取这两个文件，可直接得到会员与非会员的订单数据，再用matplotlib库进行可视化，我们得到图3-4-1和图3-4-2。

由图3-4-1可知，非会员订单数量占绝大多数，将近70%，未来的发展中可尽量多增加会员数量。

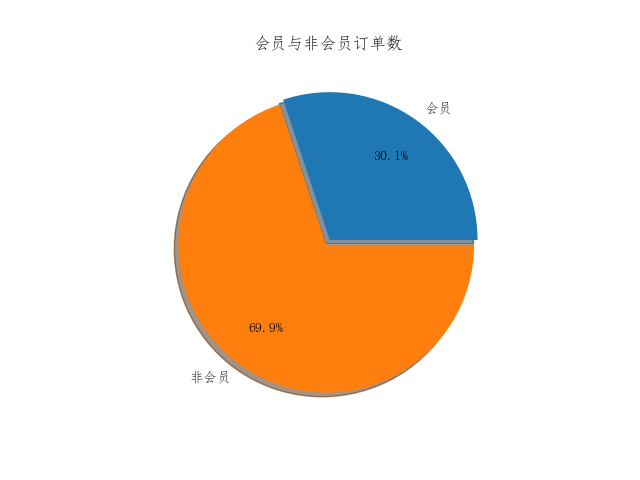


图3-4-1

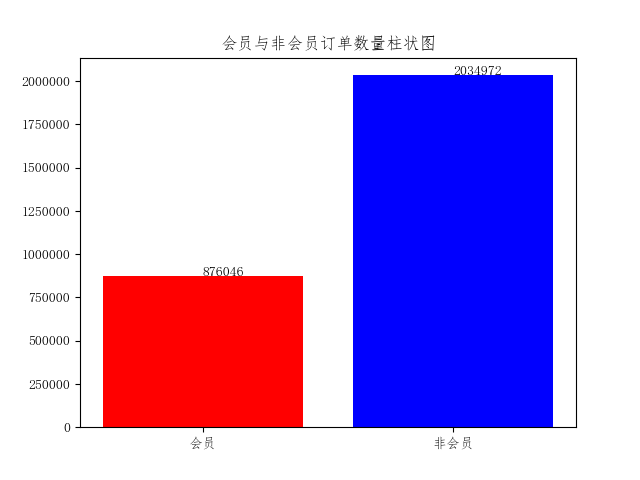


图3-4-2

## 统计会员与非会员的消费金额

会员大概率是熟客，虽然会员的订单数量低，但会员订单的消费金额比例未必就比会员订单比例低，以下对会员和非会员的消费金额进行统计。

在“会员购物情况表.csv”中，字段“消费金额”加总即可得到会员在2015年至2018年的消费金额，在“非会员购物情况表.csv”中，对字段“消费金额”进行加总即可得到非会员的2015年至2018年的消费金额。二者相加便是该商场的顾客总消费金额，得到二者数据后，用matplotlib可视化，得到图3-5-1、图3-5-2。次外，将二者除以订单数量，便可得到会员与非会员平均每订单消费金额，可视化后得到图3-5-3、图3-5-4。

由以下四图可知，虽然非会员的总消费额要比会员高，但是会员的平均每单消费额要高出很多，会员平均每单消费1338元，非会员平均每单消费797元。因此，尽量使顾客转换成会员的营销策略是有效的。

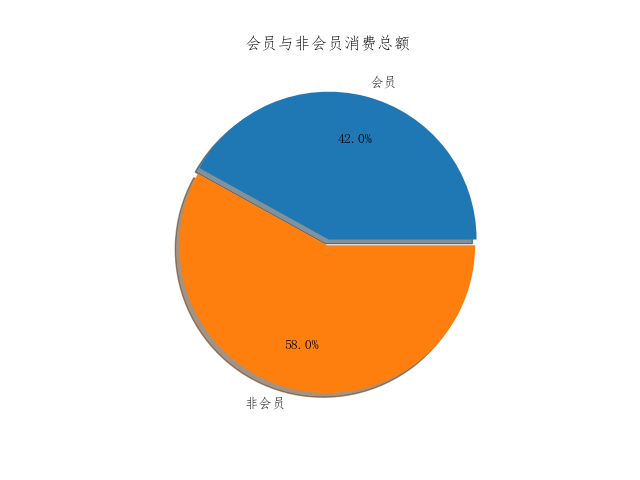


图3-5-1

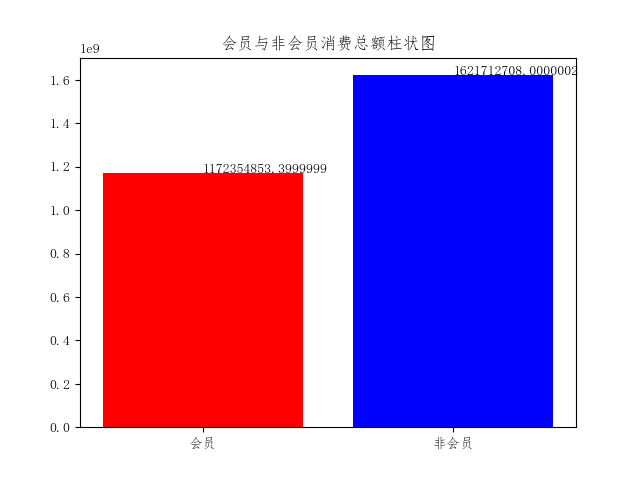


图3-5-2

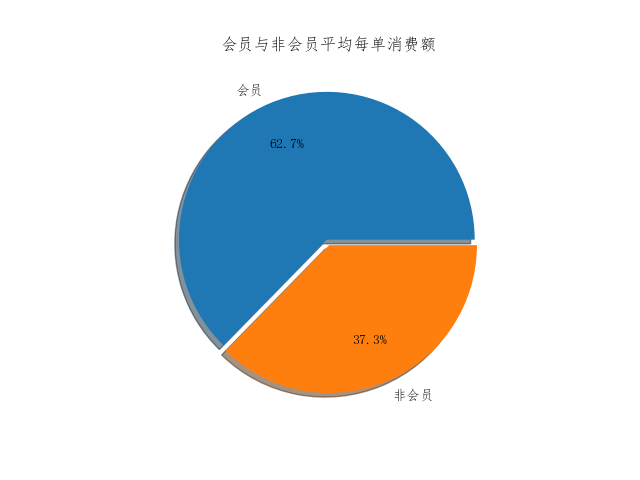


图3-5-3

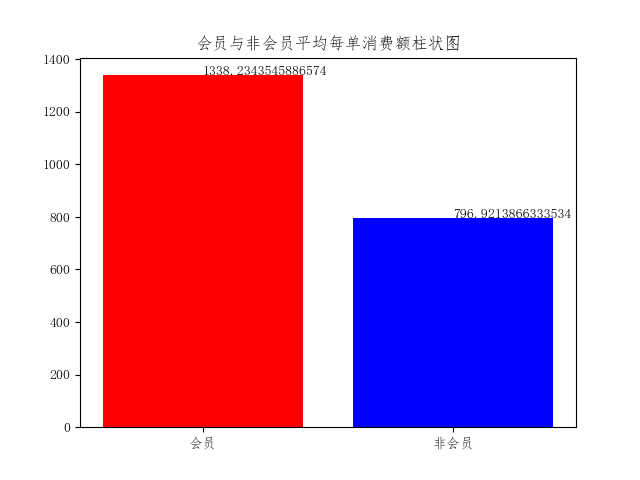


图3-5-4

## 统计不同年份的会员消费额

通过统计不同年份的会员消费额，可以直观看出会员营销策略是否有效，有效的营销策略能使得消费额得到提高，以下对会员购物情况表中的消费金额数据按年汇总。

在“会员购物情况表.csv”中，使用pandas库根据字段“消费时间”对消费金额进行汇总，可得到会员在不同年份的消费金额，使用matplotlib库进行可视化，得到图3-6-1。

从下图可直观看出，会员营销策略是成功的，会员的消费总额连年提升，其中2018年由于只有不到一个月的消费记录，因此数量上会比前3年少。

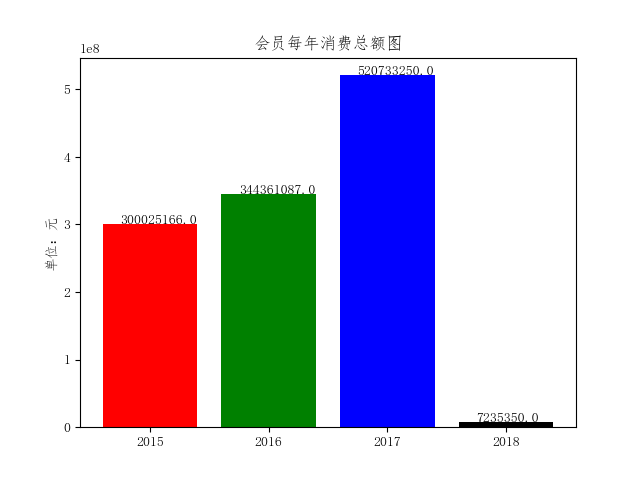


图3-6-1

## 统计不同年份的每月会员消费额

不同年份的每月消费额一定程度上能反应消费者行为在月份上的分布，以下对2015年至2018年的会员消费额进行统计。（备注：2015年的9、10、11月份数据缺失，2018年只有1月份数据）

在“会员消费情况表.csv”中，字段“消费时间”可以索引出不同年份和月份的消费记录，通过python的for循环一一将每年每月的记录索引出来，再用groupby函数进行消费总额汇总，然后使用matplotlib库进行可视化，得到图3-7-1、图3-7-2、图3-7-3、图3-7-4。

由以下四图可知，每年每月的消费总额并无明显规律，消费总额的高低起伏可能与不同时期的促销活动联系更密切。

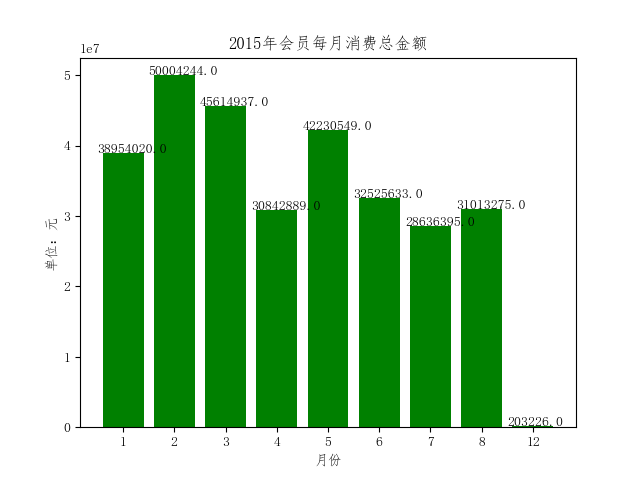


图3-7-1

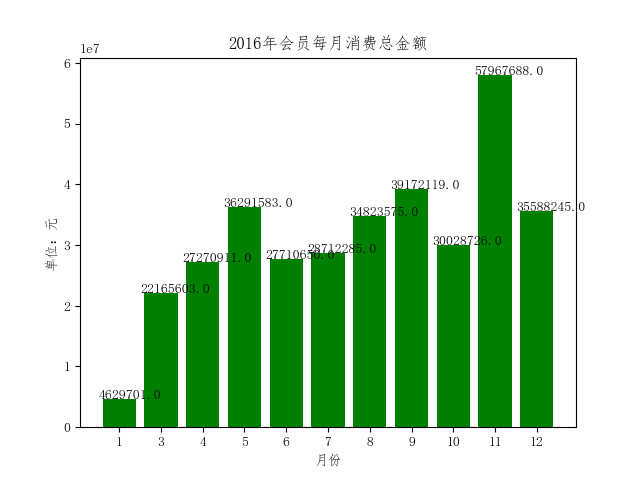


图3-7-2

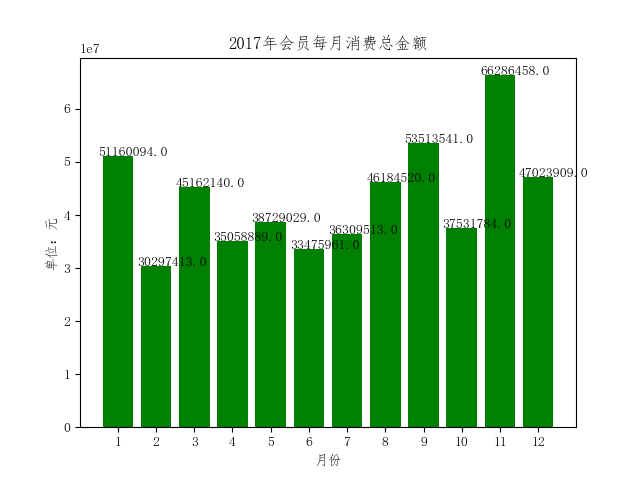


图3-7-3

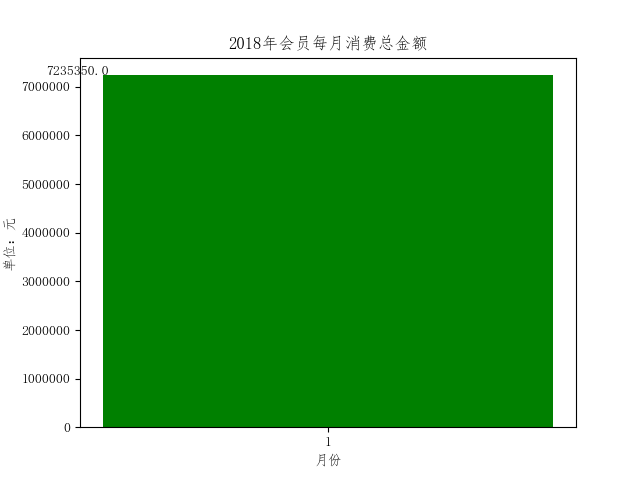


图3-7-4

## 统计一日中不同时间段的消费额

一日24小时中，不同时间段的客流量不同，因此消费金额也不同，分别统计不同时间段的消费额以及订单数量，能准确掌握消费高峰时间段，合理安排工作人员，降低经营成本，提高营销效果。以下统计不同时间段的会员消费金额，关于时间段的定义如下：

凌晨：00：00-06：00

上午：06：00-11：00

中午：11：00-14：00

下午：14：00-18：00

晚上：18：00-24：00

在“会员购物情况表.csv”中，使用pandas根据以上定义的时间段，分别索引出不同时间段的消费记录，并对字段“消费金额”进行加总，加总结果保存在列表中，然后使用matplotlib进行可视化，得到图3-8-1、图3-8-2。

由图可知，消费高峰期是下午和晚上，其次是中午。会员在下午的消费额占整天的42.39%，晚上的消费额占整天的30.9%，占比最低的时间段是凌晨和上午，凌晨消费额占比仅为0.0448%，上午的为3.66%。因此营销活动最好在中午开始进行，进行到晚上停止，在下午要加派工作人员，在上午和凌晨可少安排工作人员。

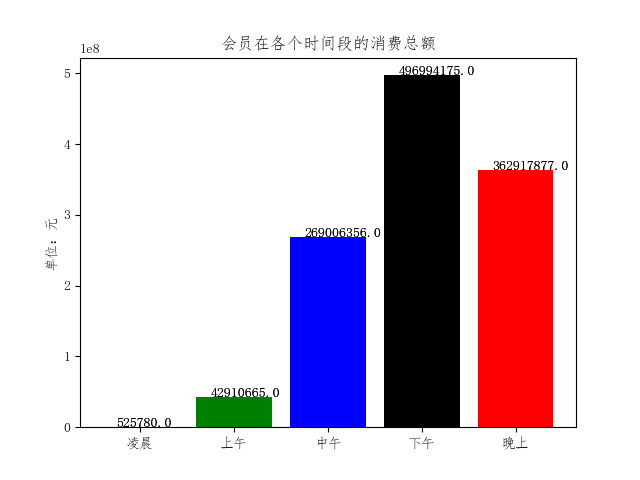


图3-8-1

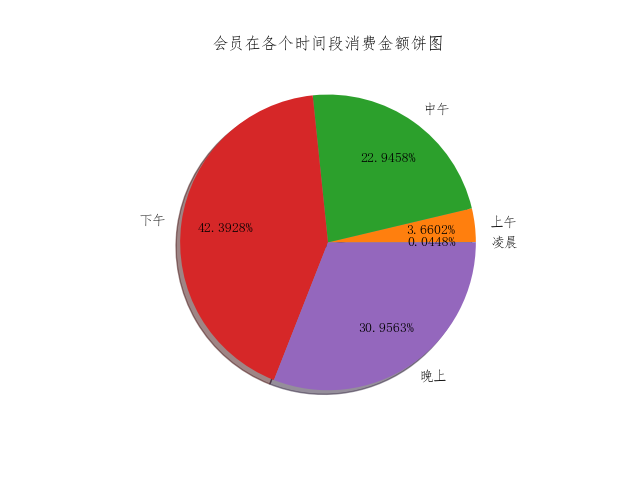


图3-8-2

## 统计每年各个季度的会员消费人次

消费者的需求可能会随着季节变化而变化，如果这一说法属实，那这种随季节变化的消费需求会直接地体现在各个季度的消费人次中，以下对每年各个季节的消费人次进行统计分析。春季为3、4、5月，夏季为6、7、8月，秋季为9、10、11月，冬季为12、1、2月，其中2015年的秋季数据缺失，2018年只有一个月因此不做统计。

在“会员购物情况表.csv”中，按字段“消费时间”索引出消费记录的发生月份，用for循环并把不同季节的月份的消费人次加总，保存在字典中，然后用matplotlib进行可视化，得到图3-9-1、图3-9-2、图3-9-3、图3-9-4。

从以下3图可看出，会员购物规律并不体现在季节中，进行营销活动时除非必要不然不必太过考虑季节，与季节有关的商品除外。

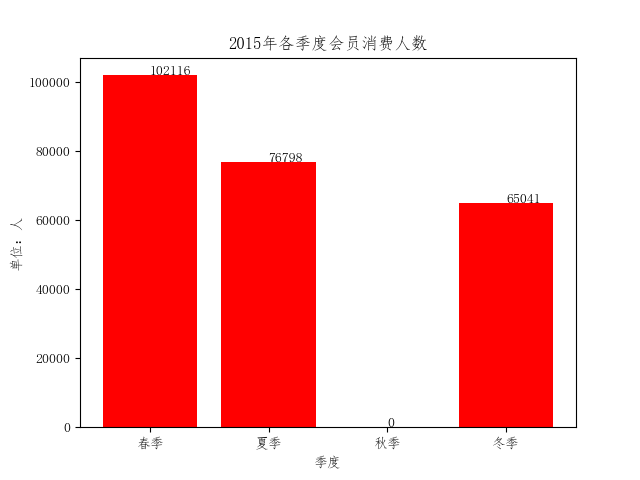


图3-9-1

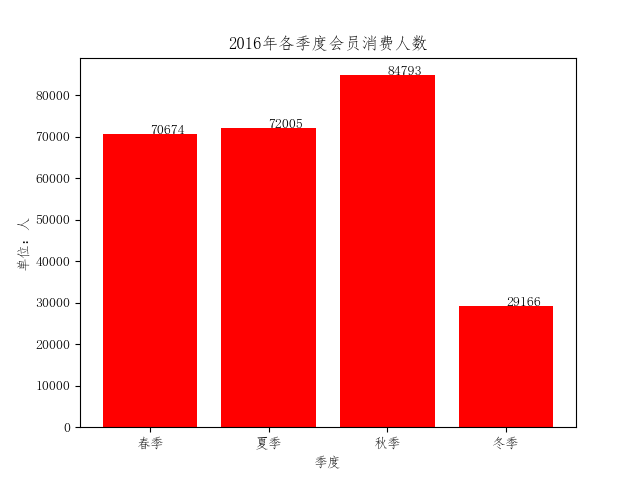


图3-9-2

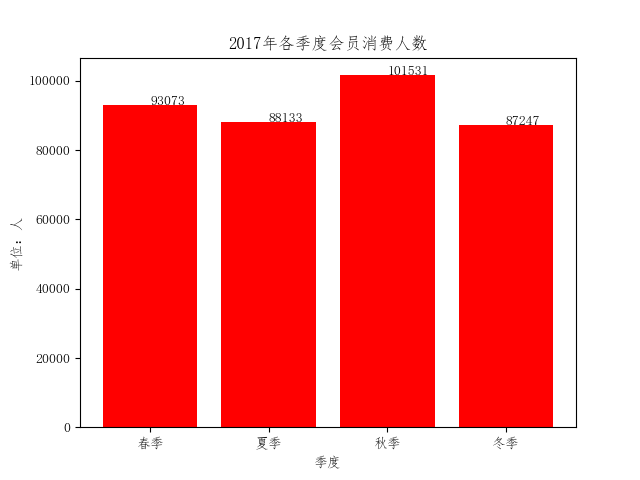


图3-9-3

## 任务实现

### 按年龄段区分会员源代码

|  |
| --- |
| '''  将会员按年龄分为青年人、中年人、老年人，并绘制饼状图并保存  44岁以下为青年人，44岁至59岁为中年人，60岁以上为老年人  '''  import pandas as pd  import matplotlib.pyplot as plt  #读取数据  data = pd.read\_csv('..\\..\\data\\有出生日期的会员信息表.csv')  data['出生日期'] = pd.to\_datetime(data['出生日期'])  #年龄构成饼图, 44岁以下为青年人，44岁至59岁为中年人，60岁以上为老年人  plt.rcParams['font.sans-serif'] = ['FangSong']  plt.figure()  plt.pie([data[2018-data['出生日期'].dt.year <= 44].shape[0],  data[(2018-data['出生日期'].dt.year > 44) & \  (2018-data['出生日期'].dt.year < 60)].shape[0],  data[2018-data['出生日期'].dt.year >= 44].shape[0]],  labels=['青年人','中年人','老年人'],  explode=[0.01,0.01,0.01],  autopct='%.1f%%',  pctdistance=0.7,)  plt.title('会员年龄占比图')  plt.savefig('..\\..\\figures\\会员年龄占比图.png')  plt.show() |

**\*代码详见：D:\python\_files\百货商场数据分析\项目任务\code\task\_2\task2-1\_1.py\***

### 按统计不同年龄段的消费金额源代码

|  |
| --- |
| '''  统计不同年龄段的消费总额，并绘制饼状图、折线图  '''  import pandas as pd  import matplotlib.pyplot as plt  #读取数据  data = pd.read\_csv('D:\\python\_files\\百货商场数据分析\\项目任务\data\\task1.csv',low\_memory=False)  #导出年龄数据  data['出生日期'] = pd.to\_datetime(data['出生日期'])  data['消费时间'] = pd.to\_datetime(data['消费时间'])  data = data[data['出生日期'].dt.year >= 1900]  data['年龄'] = 2018 - data['出生日期'].dt.year  #导出不同年龄的消费总额  data1 = data[['年龄','消费金额']].groupby(by='年龄').sum()  #作不同年龄的消费金额图  plt.rcParams['font.sans-serif'] = ['FangSong']  x = [data1.iloc[:45,0].sum(),  data1.iloc[45:60,0].sum(),  data1.iloc[60:,0].sum()]  plt.figure()  plt.pie(x,  labels=['青年人','中年人','老年人'],  explode=[0.01,0.01,0.01],  autopct='%.1f%%',  pctdistance=0.7,)  plt.title('各年龄群消费金额图')  plt.savefig('../../figures/各年龄消费金额占比图.png')  plt.show()  #各个具体年龄消费金额图  plt.figure()  data1.plot()  plt.xlabel('年龄')  plt.ylabel('消费总金额')  plt.title('各具体年龄消费总额图')  plt.savefig('../../figures//各具体年龄消费总额图.png')  plt.show() |

**\*代码详见：D:\python\_files\百货商场数据分析\项目任务\code\task\_2\task2-1\_2.py\***

### 统计不同性别的会员的人数和消费金额源代码

|  |
| --- |
| '''  统计不同性别的消费金额比例和人数比例，并绘制饼状图、柱状图  '''  import pandas as pd  import numpy as np  import matplotlib.pyplot as plt  #读取数据  data = pd.read\_csv('D:\\python\_files\\百货商场数据分析\\项目任务\data\\task1.csv',low\_memory=False)  data1 = pd.read\_csv('../../data/会员信息表.csv')  #会员性别比例图  rito = [data1[data1['性别']==1].shape[0],  data1[data1['性别']==0].shape[0]]  plt.rcParams['font.sans-serif'] = ['FangSong']  plt.figure()  plt.bar([1,2],rito,color=['r','g'],width=0.5)  plt.xticks([1,2],['男','女'])  plt.text(1,rito[0],str(rito[0]))  plt.text(2,rito[1],str(rito[1]))  plt.text(1,0.5\*rito[0],str(round(rito[0]/sum(rito),2)))  plt.text(2,0.5\*rito[1],str(round(rito[1]/sum(rito),2)))  plt.title('男女人数对比图')  plt.savefig('../../figures/男女人数对比图.png')  plt.show()  plt.figure()  plt.pie(rito[::-1],  labels=['女人','男人'],  explode=[0.01,0.01],  autopct='%.1f%%',  pctdistance=0.7,)  plt.title('男女人数占比')  plt.savefig('../../figures/男女人数占比.png')  plt.show()  #男女消费总金额对比  data2 = data[['性别','消费金额']].groupby(by='性别').sum()  plt.figure()  plt.pie(data2.values,  labels=['女人','男人'],  explode=[0.01,0.01],  autopct='%.1f%%',  pctdistance=0.7,)  plt.title('男女消费总金额占比')  plt.savefig('../../figures/男女消费总金额占比.png')  plt.show() |

**\*代码详见：D:\python\_files\百货商场数据分析\项目任务\code\task\_2\task2-1\_3.py\***

### 统计会员与非会员订单数量源代码

|  |
| --- |
| '''  根据单据号字符，分别统计会员与非会员的订单数，并对结果进行可视化，  绘制了会员与非会员的订单数量饼状图、柱状图  '''  import pandas as pd  import matplotlib.pyplot as plt  #数据读取  data1 = pd.read\_csv('../../data/会员购物情况表.csv')  data2 = pd.read\_csv('../../data/非会员购物情况表.csv')  #绘饼状图  x = [data1.shape[0],data2.shape[0]]  plt.rcParams['font.sans-serif'] = ['FangSong']  plt.figure()  plt.pie(x,  labels=['会员','非会员'],  explode=[0.02,0.02],  shadow=True,  autopct='%.1f%%',  pctdistance=0.7)  plt.title('会员与非会员订单数')  plt.savefig('../../figures/会员与非会员订单数量.png')  plt.show()  #画柱状图  plt.figure()  plt.bar(range(2),x,color=['r','b'])  plt.xticks(range(2),['会员','非会员'])  for i in range(2):  plt.text(i,x[i],str(x[i]))  plt.title('会员与非会员订单数量柱状图')  plt.savefig('../../figures/会员与非会员订单数量柱状图.png')  plt.show() |

**\*代码详见：D:\python\_files\百货商场数据分析\项目任务\code\task\_2\task2-2\_1.py\***

### 统计会员与非会员消费金额源代码

|  |
| --- |
| '''  绘制了会员与非会员的消费总额饼状图、柱状图  绘制了会员与非会员平均每单消费额柱状图  '''  import pandas as pd  import matplotlib.pyplot as plt  #数据读取  data1 = pd.read\_csv('../../data/会员购物情况表.csv')  data2 = pd.read\_csv('../../data/非会员购物情况表.csv')  #画消费总额饼状图  x = [data1['消费金额'].sum(),data2['消费金额'].sum()]  plt.rcParams['font.sans-serif'] = ['FangSong']  plt.figure()  plt.pie(x,  labels=['会员','非会员'],  explode=[0.02,0.02],  shadow=True,  autopct='%.1f%%',  pctdistance=0.7)  plt.title('会员与非会员消费总额')  plt.savefig('../../figures/会员与非会员消费总额.png')  plt.show()  #画消费总额柱状图  plt.figure()  plt.bar(range(2),x,color=['r','b'])  plt.xticks(range(2),['会员','非会员'])  for i in range(2):  plt.text(i,x[i],str(x[i]))  plt.title('会员与非会员消费总额柱状图')  plt.savefig('../../figures/会员与非会员消费总额柱状图.png')  plt.show()  #画平均每单消费额饼状图  x = [data1['消费金额'].mean(),data2['消费金额'].mean()]  plt.rcParams['font.sans-serif'] = ['FangSong']  plt.figure()  plt.pie(x,  labels=['会员','非会员'],  explode=[0.02,0.02],  shadow=True,  autopct='%.1f%%',  pctdistance=0.7)  plt.title('会员与非会员平均每单消费额')  plt.savefig('../../figures/会员与非会员平局每单消费额.png')  plt.show()  #画平均每单消费额柱状图  plt.figure()  plt.bar(range(2),x,color=['r','b'])  plt.xticks(range(2),['会员','非会员'])  for i in range(2):  plt.text(i,x[i],str(x[i]))  plt.title('会员与非会员平均每单消费额柱状图')  plt.savefig('../../figures/会员与非会员平均每单消费额柱状图.png')  plt.show() |

**\*代码详见：D:\python\_files\百货商场数据分析\项目任务\code\task\_2\task2-2\_2.py\***

### 统计不同年份的会员消费额源代码

|  |
| --- |
| '''  绘制了不同年份的会员消费总额的柱状图  '''  import pandas as pd  import matplotlib.pyplot as plt  #读取文件  data = pd.read\_csv('../../data/会员购物情况表.csv')  data['消费时间'] = pd.to\_datetime(data['消费时间'])  year = data['消费时间'].dt.year.unique()  x = []  for i in range(len(year)):  x.append(data.loc[data['消费时间'].dt.year==year[i],'消费金额'].sum())  #绘制条形图  plt.rcParams['font.sans-serif'] = ['FangSong']  plt.bar(range(4),x,color=list('rgbk'))  plt.xticks(range(4),year)  for i in range(4):  plt.text(i-0.25,x[i],round(x[i]))  plt.ylabel('单位：元')  plt.title('会员每年消费总额图')  plt.savefig('../../figures/会员每年消费总额图.png')  plt.show() |

**\*代码详见：D:\python\_files\百货商场数据分析\项目任务\code\task\_2\task2-2\_3.py\***

### 统计不同年份会员每月消费额源代码

|  |
| --- |
| '''  按不同年份绘制了每个月的消费总额柱状图  '''  import pandas as pd  import matplotlib.pyplot as plt  data = pd.read\_csv('../../data/会员购物情况表.csv')  data['消费时间'] = pd.to\_datetime(data['消费时间'])  plt.rcParams['font.sans-serif'] = ['FangSong']  def plot\_figure(x,year,xticks):  plt.figure()  plt.bar(range(len(x)),  x,  color='g')  plt.title(str(year)+'年会员每月消费总金额')  plt.ylabel('单位：元')  plt.xlabel('月份')  plt.xticks(range(len(x)),xticks)  for i in range(len(x)):  plt.text(i-0.5,x[i],round(x[i]))  filename = '../../figures/' + str(year)+'年会员每月消费总金额.png'  plt.savefig(filename)  plt.show()  year = data['消费时间'].dt.year.unique()  month\_data = {}  for i in range(len(year)):  df = data[data['消费时间'].dt.year==year[i]]  months = df['消费时间'].dt.month.unique()  month\_list = []  for j in range(len(months)):  month\_list.append(df.loc[df['消费时间'].dt.month==months[j],'消费金额'].sum())  month\_data[year[i]] = [month\_list,months]  for y in year:  plot\_figure(x=month\_data[y][0],year=y,xticks=month\_data[y][1]) |

**\*代码详见：D:\python\_files\百货商场数据分析\项目任务\code\task\_2\task2-2\_4.py\***

### 统计一日中不同时间段的消费额源代码

|  |
| --- |
| '''  把一天划分为以下几个时间段  凌晨：00：00-06：00  上午：06：00-11：00  中午：11：00-14：00  下午：14：00-18：00  晚上：18：00-24：00  并对会员再每个时间段的消费总额进行了统计，  根据统计结果绘制了柱状图、饼状图  '''  import pandas as pd  import matplotlib.pyplot as plt  #读取数据  data = pd.read\_csv('../../data/会员购物情况表.csv')  data['消费时间'] = pd.to\_datetime(data['消费时间'])  #各时间段消费金额提取  x = []  x.append(data.loc[data['消费时间'].dt.hour < 6,'消费金额'].sum())  x.append(data.loc[(data['消费时间'].dt.hour>=6) & (data['消费时间'].dt.hour<11),'消费金额'].sum())  x.append(data.loc[(data['消费时间'].dt.hour>=11) & (data['消费时间'].dt.hour<14),'消费金额'].sum())  x.append(data.loc[(data['消费时间'].dt.hour>=14) & (data['消费时间'].dt.hour<18),'消费金额'].sum())  x.append(data.loc[(data['消费时间'].dt.hour>=18) & (data['消费时间'].dt.hour<=23),'消费金额'].sum())  #绘制柱状图  plt.rcParams['font.sans-serif'] = ['FangSong']  plt.bar(x=range(5),height=x,color=list('rgbkr'))  for i in range(5):  plt.text(i-0.25,x[i],round(x[i]))  plt.xticks(range(5),['凌晨','上午','中午','下午','晚上'])  plt.title('会员在各个时间段的消费总额')  plt.ylabel('单位：元')  plt.savefig('../../figures/会员在各个时间段的消费金额柱状图.png')  plt.show()  #绘制饼状图  plt.figure()  plt.pie(x=x,  labels=['凌晨','上午','中午','下午','晚上'],  shadow=True,  autopct='%.4f%%',  pctdistance=0.7)  plt.title('会员在各个时间段消费金额饼图')  plt.savefig('../../figures/会员在各个时间段消费金额饼图.png')  plt.show() |

**\*代码详见：D:\python\_files\百货商场数据分析\项目任务\code\task\_2\task2-3\_1.py\***

### 统计每年各个季度的会员消费人次源代码

|  |
| --- |
| '''  把一年分为春夏秋冬四个季节，分别统计会员在各年的各个季节中的消费总额，  并根据季节和年份绘制了柱状图  '''  import pandas as pd  import matplotlib.pyplot as plt  #读取数据  data = pd.read\_csv('../../data/会员购物情况表.csv')  data['消费时间'] = pd.to\_datetime(data['消费时间'])  #定义绘图函数  plt.rcParams['font.sans-serif'] = ['FangSong']  def plot\_figure(x,year,xticks):  plt.figure()  plt.bar(range(len(x)),  x,  color='r')  plt.title(str(year)+'年各季度会员消费人数')  plt.ylabel('单位：人')  plt.xlabel('季度')  plt.xticks(range(len(x)),xticks)  for i in range(len(x)):  plt.text(i,x[i],round(x[i]))  filename = '../../figures/' + str(year)+'年各季度会员消费人数.png'  plt.savefig(filename)  plt.show()  years = [2015,2016,2017]  season\_dict = {}  for year in years:  df = data[data['消费时间'].dt.year==year]  x = []  x.append(df[(df['消费时间'].dt.month==3) | \  (df['消费时间'].dt.month==4) | \  (df['消费时间'].dt.month==5)].shape[0])  x.append(df[(df['消费时间'].dt.month == 6) | \  (df['消费时间'].dt.month == 7) | \  (df['消费时间'].dt.month == 8)].shape[0])  x.append(df[(df['消费时间'].dt.month == 9) | \  (df['消费时间'].dt.month == 10) | \  (df['消费时间'].dt.month == 11)].shape[0])  x.append(df[(df['消费时间'].dt.month == 12) | \  (df['消费时间'].dt.month == 1) | \  (df['消费时间'].dt.month == 2)].shape[0])  season\_dict[year] = x  #绘图并保存  for year in years:  plot\_figure(x=season\_dict[year],  year=year,  xticks=['春季','夏季','秋季','冬季']) |

**\*代码详见：D:\python\_files\百货商场数据分析\项目任务\code\task\_2\task2-3\_2.py\***

# 会员用户画像

**任务描述**

用户画像是一种勾画目标用户、联系用户诉求与设计方向的有效工具。在本次案例中，用户画像指的是根据会员的基本特征（如年龄性别等）、消费习惯特征（如消费时间段）、消费偏好特征（购买的商品等）来刻画会员的消费行为，以便更好地进行营销活动。

**任务分析**

会员用户画像可以分为以下4个步骤。

1. 构建会员基本特征
2. 构建会员业务特征
3. 构建会员兴趣特征
4. 会员特征可视化

## 构建会员基本特征

会员的基本特征有年龄、性别、入会时间等等，除了这些可直接从会员信息表中得到的特征，还可以由基本特征衍生出一些更利于分析的特征。本案例构建的基本特征由年龄段（青年人、中年人、老年人）、性别、入会时长、总消费金额。

使用pandas读入“task1.csv”和“会员信息表.csv”，会员信息表中可直接得到基本特征中的“性别”特征；由2018年减去会员的出生年份可得到会员年龄，进而根据年龄可给会员贴上青年人（45岁以下）、中年人（45岁至60岁）、老年人（60岁以上）的标签，进而得到基本特征中的“年龄段”特征；用2018减去入会年份，即可得到基本特征中的“入会时长”；在“task1.csv”文件中，用pandas的groupby函数把会员卡号与消费金额汇总，再把汇总结果与以上得到的3个基本特征的dataframe合并，即可得到会员的消费总金额特征。所构建的会员基本特征保存在“task3\_1.csv”文件中，部分信息如表4-1-1。

表4-1-1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 会员卡号 | 出生日期 | 性别 | 入会时间 | 年龄段 | 入会时长 | 消费金额 |
| 8fd77c69 | 1978-10-14 00:00:00.000 | 女 | 2002-02-25 00:00:00.000 | 青年人 | 16 | 0.0 |
| 393285c3 | 1968-05-09 00:00:00.000 | 女 | 2002-10-23 21:10:35.260 | 中年人 | 16 | 9888.96 |
| ba52e695 | 1977-05-18 00:00:00.000 | 女 | 2002-10-24 11:06:39.980 | 青年人 | 16 | 0.0 |
| c497071b | 1970-08-14 00:00:00.000 | 女 | 2002-10-24 11:27:08.440 | 中年人 | 16 | 0.0 |
| ee68a6de | 1974-09-20 00:00:00.000 | 女 | 2002-10-24 11:31:38.130 | 青年人 | 16 | 0.0 |

## 构建会员业务特征

本案例中，根据会员的业务而构建的业务特征有“消费水平”、“新老会员”、“总消费次数”、“购买过的商品”。其中总消费金额在300元以下的定义为低消费，总消费金额在300到1500之间的定义为中等消费，总消费金额在1500以上的定义为高消费；入会时间5年以下的定义为新会员，入会时间5年以上的定义为老会员；总消费次数指的是会员在2015年至2018年的消费记录次数，不使用会员卡的不计算在内；购买过的商品指的是会员在2015年至2018年所购买过的所有商品名称列表。

使用pandas库读入“task1.csv”和“task3\_1.csv”，特征“消费水平”可用pandas的pd.cut()函数对“task3\_1.csv”中的字段“消费金额”进行切分得到；特征“新老会员”同样通过pd.cut()函数对“task3\_1.csv”中的字段“入会时长”进行切分得到；特征“总消费次数”的获取，先使用pandas的groupby函数对会员卡号及消费金额记录进行次数统计，再使用pd.merge()与“task3\_1.csv”合并，便可得到；特征“购买过的商品”的获取，先用groupby函数以及agg()函数对“task1.csv”的会员卡号和商品名称字段进行统计，聚合函数为np.array，便可得到。构建的会员业务特征保存在文件“task3\_2.csv”中，部分数据如表4-2-1。

表4-2-1

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 会员卡号 | 性别 | 年龄段 | 入会时长 | 消费金额\_x | 消费水平 | 新老会员 | 消费次数 | 购买过的商品 |
| 8fd77c69 | 女 | 青年人 | 16 | 0.0 | 低消费 | 老会员 | 0 |  |
| 393285c3 | 女 | 中年人 | 16 | 9888.96 | 高消费 | 老会员 | 7 | "['VME女鞋系列C双' 'VME女鞋系列D双' 'LIU JOF件' 'Thursday Island A无' '例外 B.5件' 'CAGLI WASH A件' 'N.PAIA 正价件']" |

## 构建会员兴趣特征

本案例构建的会员兴趣特征有“购物季节”、“购物时间段”。“购物季节”有春天、夏天、秋天、冬天，是由会员的所有购物记录中出现最多的季节得来，“购物时间段”有凌晨、上午、中午、下午、晚上，是由会员所有购物记录中出现最多的时间段得来。掌握了会员最常出现购物行为的时间段和季节，可以更精准地对其进行营销，比如在会员即将购物前发放促销商品的优惠券等。

使用pandas打开“会员购物情况表.csv”和“task3\_2.csv”，用pd.cut()对字段“消费时间”的月份进行操作，提取字段“购物季节”和“会员卡号”出来再使用groupby()函数进行汇总，汇总方式为max()，汇总结果使用pd.merge()与“task3\_2.csv”合并，便可得到特征“购物季节”。应用类似的操作对字段“消费时间”的小时进行操作可得到特征“购物时间段”。得到两个会员兴趣特征后，将其保存在文件“task3\_3.csv”，部分数据信息如表4-3-1。备注：若有会员入会后从未消费，则其在某些特征上会存在缺失值。

表4-3-1

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 会员卡号 | 性别 | 年龄段 | 入会时长 | 消费金额\_x | 消费水平 | 新老会员 | 消费次数 | 购买过的商品 | 季节 | 购物时间段 |
| 8fd77c69 | 女 | 青年人 | 16 | 0.0 | 低消费 | 老会员 | 0 |  |  |  |
| 393285c3 | 女 | 中年人 | 16 | 9888.96 | 高消费 | 老会员 | 7 | "['VME女鞋系列C双' 'VME女鞋系列D双' 'LIU JOF件' 'Thursday Island A无' '例外 B.5件' 'CAGLI WASH A件' 'N.PAIA 正价件']" | 夏天 | 晚上 |

## 会员特征可视化

会员画像的步骤已经完成，为了更直观地呈现画像效果，现在随机选取一名会员，对其特征进行可视化。本案例选的是会员卡号为“8e08deed”的会员。

使用pandas从“task3\_3.csv”中提取出该会员的特征后，先用list()把特征化为列表，再用join()方法把列表变成字符串，再使用WordCloud库进行词云图绘制。得到图4-4-1。

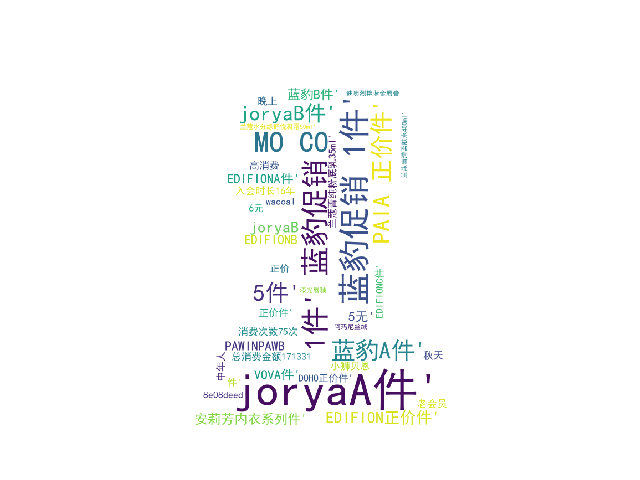


图4-4-1

## 任务实现

### 构建会员基本特征源代码

|  |
| --- |
| '''  构建会员基本特征，主要有如下  年龄段：[青年人，中年人，老年人]；  性别：[男，女]  入会时长：0，1，2，...  消费金额：  添加的特征保存在“task3\_1.csv”  '''  import pandas as pd  import numpy as np  task1 = pd.read\_csv('../../data/task1.csv',low\_memory=False)  data = pd.read\_csv('../../data/会员信息表.csv')  #性别特征  data['性别'] = data['性别'].map({1.0:'男',0.0:'女'})  #年龄段  data['出生日期'] = pd.to\_datetime(data['出生日期'])  data['年龄段'] = '未知'  data.loc[(2018-data['出生日期'].dt.year<=44)&(2018-data['出生日期'].dt.year>=0),'年龄段'] = '青年人'  data.loc[(2018-data['出生日期'].dt.year<=59)&(2018-data['出生日期'].dt.year>=45),'年龄段'] = '中年人'  data.loc[(2018-data['出生日期'].dt.year<=100)&(2018-data['出生日期'].dt.year>=60),'年龄段'] = '老年人'  data.drop(data[data['年龄段']=='未知'].index, inplace=True)  #入会时长  data['入会时间'] = pd.to\_datetime(data['入会时间'])  data['入会时长'] = 2018 - data['入会时间'].dt.year  #消费金额  cumsume = task1[['会员卡号','消费金额']].groupby(by='会员卡号').sum()  data = pd.merge(data,cumsume,on='会员卡号',how='inner')  data.to\_csv('../../data/task3\_1.csv',index=False) |

**\*代码详见：D:\python\_files\百货商场数据分析\项目任务\code\task\_3\task3-1\_1.py \***

### 构建会员业务特征源代码

|  |
| --- |
| '''  根据业务特征构建的特征标签有：  消费水平：[低消费，中等消费，高消费]  新老会员：[新会员，老会员]  总消费次数：int  购买过的商品：  添加的特征保存在“task3\_2.csv”  '''  import pandas as pd  import numpy as np  task1 = pd.read\_csv('../../data/task1.csv',low\_memory=False)  data = pd.read\_csv('../../data/task3\_1.csv')  #消费水平  data['消费水平'] = pd.cut(data['消费金额'],  bins=[-0.001,300,1500,10000000000],  labels=['低消费','中等消费','高消费'])  #新老会员  data['新老会员'] = pd.cut(data['入会时长'],  bins=[-0.1,5,17],  labels=['新会员','老会员'])  #总消费次数  count = task1[['会员卡号','消费金额']].groupby(by='会员卡号').count()  data = pd.merge(data,count,how='inner',on='会员卡号')  data.rename({'消费金额\_y':'消费次数'},axis=1,inplace=True)  #购买过的商品  df = task1[['会员卡号','商品名称']].groupby(by='会员卡号').\  agg(购买过的商品=pd.NamedAgg(column='商品名称', aggfunc=np.array))  data = pd.merge(data,df,on='会员卡号',how='inner')  #删除无关特征  data.drop(['出生日期','入会时间'],axis=1,inplace=True)  #保存处理结果  data.to\_csv('../../data/task3\_2.csv',index=False) |

**\*代码详见：D:\python\_files\百货商场数据分析\项目任务\code\task\_3\task3-1\_2.py \***

### 构建会员兴趣特征源代码

|  |
| --- |
| '''  根据会员用户的兴趣特征，构建如下字段：  购物季节：春天、夏天、秋天、冬天  购物时间段：凌晨、上午、中午、下午、晚上  构建的特征保存在“task3\_3.csv”文件中  '''  import pandas as pd  data = pd.read\_csv('../../data/task3\_2.csv')  task1 = pd.read\_csv('../../data/会员购物情况表.csv',low\_memory=False)  print(data.shape) #(142701, 9)  print(task1.shape) #(1013660, 12)  #购物季节  task1['消费时间'] = pd.to\_datetime(task1['消费时间'])  task1['季节'] = pd.cut(task1['消费时间'].dt.month,  bins=[0,2,5,8,11,13],  labels=['冬天','春天','夏天','秋天','冬'])  task1.loc[task1['季节']=='冬','季节'] = '冬天'  df = task1[['会员卡号','季节']].groupby(by='会员卡号').max()  data = pd.merge(data,df,on='会员卡号',how='left')  #购物时间段  task1['购物时间段'] = pd.cut(task1['消费时间'].dt.hour,  bins=[0,6,11,14,18,24],  labels=['凌晨','上午','中午','下午','晚上'])  task1.drop(task1[task1['购物时间段'].isnull().values==True].index,inplace=True)  df1 = task1[['会员卡号','购物时间段']].groupby(by='会员卡号').max()  data = pd.merge(data,df1,on='会员卡号',how='left')  #保存  data.to\_csv('../../data/task3\_3.csv',index=False) |

**\*代码详见：D:\python\_files\百货商场数据分析\项目任务\code\task\_3\task3-1\_3.py\***

### 会员特征可视化源代码

|  |
| --- |
| '''  随机选取一名会员进行画像，以词云图展示  '''  import pandas as pd  from wordcloud import WordCloud  from scipy.misc import imread  import matplotlib.pyplot as plt  data = pd.read\_csv('../../data/task3\_3.csv')  #选取一名会员并将其信息存到列表customer中  customer = list(data.iloc[19,:].values)  merchandise = customer.pop(-3)  merchandise = merchandise[2:-2].split()  customer.extend(merchandise)  print(customer)  #绘制词云图  ##文本处理  customer[3] = '入会时长' + str(customer[3]) + '年'  customer[4] = '总消费金额' + str(customer[4]) + '元'  customer[7] = '消费次数' + str(customer[7]) + '次'  text = ' '.join(customer)  ##创建词云对象  back\_figure = imread('../../figures/词云底图2.jpg')  wordcloud = WordCloud(background\_color='white',  width=2000,  height=1000,  margin=2,  max\_words=100,  mask=back\_figure,  font\_path='simhei.ttf',  random\_state=10)  w = wordcloud.generate(text) # 传入需画词云图的文本  plt.imshow(w)  plt.axis('off')  plt.savefig('../../figures/词云图2.png')  plt.show() |

**\*代码详见：D:\python\_files\百货商场数据分析\项目任务\code\task\_3\task3-1\_4.py\***

# 会员用户细分及营销方案制定

**任务描述**

以上构建的会员特征能直接体现会员差异，但不同会员可能千差万别，这对营销方案的制定带来了困难。为了更好地制定营销方案，还需要对会员用户进行进一步的细分。要对会员用户进行细分，首先要对会员的特征进行价值分析，直接提取或构建新的价值特征，然后对价值特征进行聚类建模，分出不同的类，再对不同的会员类有针对地制定营销方案。

**任务分析**

对会员细分及制定营销方案可以分为以下3个步骤。

1. 构建会员价值分析特征指标
2. 使用K-Means聚类
3. 营销方案制定

## 构建会员价值分析特征指标

本案例构建的会员特征有“Recency”，“Monetary”，“Frequency”。Recency指的是最近一次消费时间与截止时间的间隔。通常情况下，最近一次消费时间与截止时间的间隔越短，对即时提供的商品或是服务也最有可能感兴趣。Frequency指顾客在某段时间内所消费的次数。可以说消费频率越高的顾客，也是满意度越高的顾客，其忠诚度也就越高，顾客价值也就越大。Monetary指顾客在某段时间内所消费的金额。消费金额越大的顾客，他们的消费能力自然也就越大，这就是所谓“20%的顾客贡献了80%的销售额”的二八法则。

使用pandas读取“task3\_3.csv”文件，特征“Recency”可直接用“2018-01-31”减去字段“消费时间”得到；特征“Monetary”即原字段“消费金额”；特征“Frequency”用字段“消费次数”除以3得到；价值特征数据保存在“task4\_1.csv”中，部分数据如表5-1-1。

表5-1-1

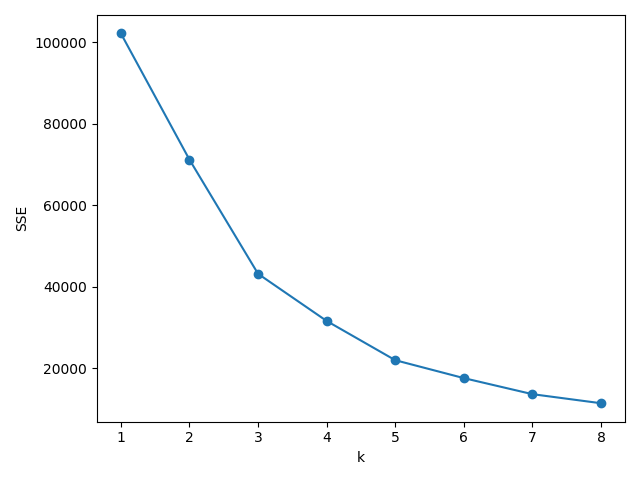
|  |  |  |  |
| --- | --- | --- | --- |
| 会员卡号 | Recency | Monetary | Frequency |
| 393285c3 | 255.0 | 9888.96 | 2.3333 |
| b956250d | 1066.0 | 11369.0 | 1.6667 |
| 523adac0 | 1102.0 | 160.0 | 0.3333 |
| daf8977f | 79.0 | 14471.0 | 2.6667 |
| 8e08deed | 30.0 | 171331.6 | 25.0 |

## K-Means聚类

使用5.1中得到的“task4\_1.csv”，结合业务需求，本案例决定用K-means算法对会员特征进行聚类建模。应用k-means算法有3个步骤，数据标准化预处理、选取k值、模型训练。

会员价值特征的量纲不同，直接应用于模型可能会导致聚类效果不好，因此要对特征数据进行预处理，使其量纲相同。本案例使用的预处理方法是数据标准化，通过sklearn的preprocessing模块的StandardScaler()函数完成。

k-means聚类算法中的k值需要预先定义，但k值的选取本身是个问题，不同的k值的聚类效果会不一样。本案例用的是肘部法则（Elbow Method），种方法适用于 K 值相对较小的情况，当选择的k值小于真正的时，k每增加1，cost值就会大幅的减小；当选择的k值大于真正的K时， k每增加1，cost值的变化就不会那么明显。这样，正确的k值就会在这个转折点，类似elbow的地方，因此k值定为3。



定好k值并将数据标准化预处理后，通过sklearn进行聚类建模，聚类结果保存在“task4\_2.csv”中，如表5-2-1。其中第0类会员有12194人，第1类会员有21147人，第2类会员有697人。

表5-2-1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 会员卡号 | Recency | Monetary | Frequency | 类别 |
| 393285c3 | 255.0 | 9888.96 | 2.3333 | 1 |
| b956250d | 1066.0 | 11369.0 | 1.6667 | 0 |
| 523adac0 | 1102.0 | 160.0 | 0.3333 | 0 |
| daf8977f | 79.0 | 14471.0 | 2.6667 | 1 |
| 8e08deed | 30.0 | 171331.6 | 25.0 | 2 |

得到聚类结果后，进一步使用matplotlib对聚类结果进行可视化，以更直观地呈现聚类效果，可视化的方式是雷达图，如图5-2-1。

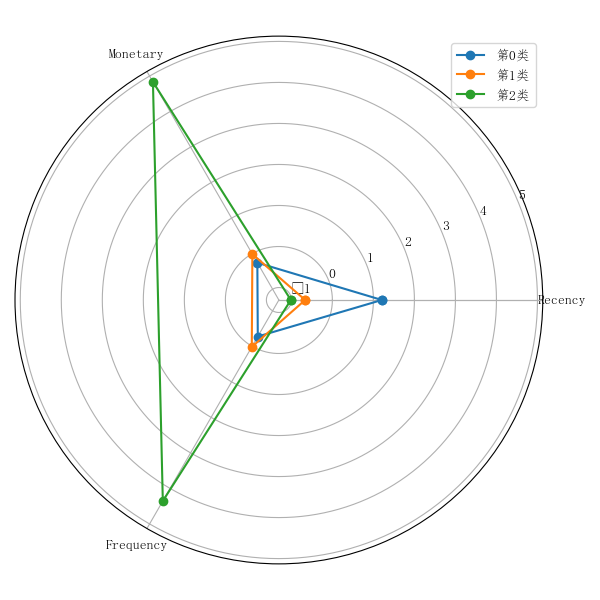


图5-2-1

## 营销方案制定

统计各类会员的人数以及消费总额，我们得到图5-3-1和图5-3-2。

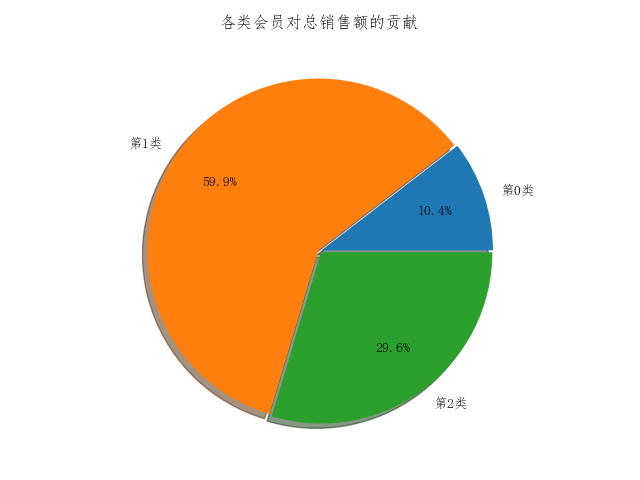
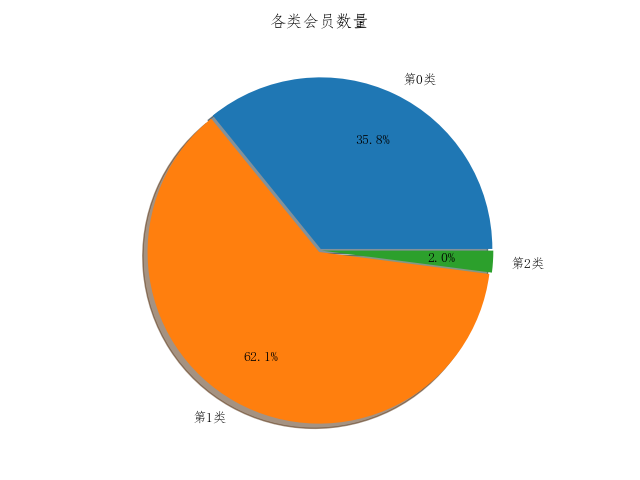


图5-3-1 图5-3-2

由以上两图可知，第0类会员数量占比35.8%，贡献的消费总额为10.4%。根据图5-2-1可知，第0类会员的特点为消费总额（Monetary）低、消费频率（Frequency）低、距离上次消费间隔（Recency）长。从营销角度看，这类会员是低价值的，消费额低往往能说明他们的购物行为更为理性，因此针对其制定的营销活动应该选择受众面比较广的商品，比如日用百货，促销服装等。

由图5-3-1和图5-3-2以及图5-2-1可知，第1类会员数量占比62.1%，消费总额贡献占比59.9%，基本与数量占比持平。第1类会员的特点为消费总额（Monetary）一般、消费频率（Frequency）一般、距离上次消费间隔（Recency）短，他们是商场的收入的最重要来源。这类会员需要得到重视，针对其制定的营销方案，应该充分考虑商品的关联性，可推荐他们曾购买的商品的周边商品，以及一些实用性较强的日用百货，或者发放一些生活必需品的中低门槛的满减优惠券。

由图5-3-1和图5-3-2和图5-2-1可知，第2类会员数量占比仅为2.0%，但其贡献的消费总额占比高达29.6%，他们的特点为消费总额（Monetary）高、消费频率（Frequency）高、距离上次消费间隔（Recency）短。毫无疑问这类会员的价值是最高的，针对其制定的营销方案应该考虑提高其忠诚度，让其成为忠诚客户，向其推荐的商品应该选择高端系列产品，突出产品的品牌、档次，向其发放的优惠券的满减门槛应足够高，甚至向其推销的工作人员的服务意识、个人形象要有严格的要求。

## 任务实现

### 构建会员价值分析特征指标源代码

|  |
| --- |
| '''  构建的价值分析特征指标主要有：  ①Recency 最后一次消费时间与截止日期的间隔  ②Frequency 每年消费频率  ③Monetary 总消费额  '''  import pandas as pd  data = pd.read\_csv('../../data/task3\_3.csv')  task1 = pd.read\_csv('../../data/task1.csv',low\_memory=False)  task1['消费时间'] = pd.to\_datetime(task1['消费时间'])  # Recency  task1['Recency'] = pd.to\_datetime('2018/01/31') - task1['消费时间']  task1['Recency'] = task1['Recency'].dt.days  df = task1[['会员卡号','Recency']].groupby(by='会员卡号').min()  df = df.dropna()  data = pd.merge(data,df,on='会员卡号')  #Monetary  data.rename({'消费金额\_x':'Monetary'},axis=1,inplace=True)  #Frequency  data['Frequency'] = round(data['消费次数'] /3,4)  data[['会员卡号','Recency','Monetary','Frequency']].to\_csv('../../data/task4\_1.csv',index=False) |

**\*代码详见：D:\python\_files\百货商场数据分析\项目任务\code\task\_4\task4-1\_1.py\***

### k-means聚类建模源代码

|  |
| --- |
| '''  使用Kmeans聚类算法，对“task4\_1.csv"文件的数据进行聚类，  并进行雷达图可视化  '''  from sklearn.preprocessing import StandardScaler  from sklearn.cluster import KMeans  import pandas as pd  import matplotlib.pyplot as plt  import numpy as np  data0 = pd.read\_csv('../../data/task4\_1.csv')  data = data0[['Recency','Monetary','Frequency']].copy()  #数据标准化预处理  sc = StandardScaler()  standard\_data = sc.fit\_transform(data)  #选取K值  SSE = []  for k in range(1,9):  estimator = KMeans(n\_clusters=k)  estimator.fit(standard\_data)  SSE.append(estimator.inertia\_)  X = range(1, 9)  plt.xlabel('k')  plt.ylabel('SSE')  plt.plot(X, SSE, 'o-')  plt.show()  #模型构建与训练  model = KMeans(n\_clusters=3,random\_state=0,max\_iter=500)  fit\_model = model.fit(standard\_data)  data0['类别'] = fit\_model.predict(standard\_data)  data0.to\_csv('../../data/task4\_2.csv',index=False)  #模型可视化  print('聚类中心：\n',fit\_model.cluster\_centers\_) #聚类中心  plt.rcParams['font.family'] = 'FangSong'  plt.rcParams['font.sans-serif'] = ['FangSong']  label = np.array(['Recency','Monetary','Frequency'])  angles = np.linspace(0,2\*np.pi,3,endpoint=False)  angles = np.concatenate((angles,[angles[0]]))  centers = fit\_model.cluster\_centers\_  plt\_data = np.concatenate((centers,centers[:,[0]]),axis=1)  fig = plt.figure(figsize=(6,6))  ax = fig.add\_subplot(111,polar=True)  for i in range(len(plt\_data)):  ax.plot(angles,plt\_data[i],'o-',label=label[i])  ax.set\_thetagrids(angles\*180/np.pi,label)  #plt.legend(bbox\_to\_anchor=(0.8,1.15),ncol=3)  plt.legend(['第0类','第1类','第2类'])  plt.savefig('../../figures/雷达图.png')  plt.show()  x = [data0.loc[data0['类别'] == 0, 'Monetary'].count(),  data0.loc[data0['类别'] == 1, 'Monetary'].count(),  data0.loc[data0['类别'] == 2, 'Monetary'].count()]  plt.pie(x, explode=[0.01,0.01,0.01],  labels=['第0类', '第1类', '第2类'],  shadow=True,  autopct='%.1f%%',  pctdistance=0.7  )  plt.title('各类会员数量')  plt.show()  y = [data0.loc[data0['类别'] == 0, 'Monetary'].sum(),  data0.loc[data0['类别'] == 1, 'Monetary'].sum(),  data0.loc[data0['类别'] == 2, 'Monetary'].sum()]  plt.pie(y, explode=[0.01,0.01,0.01],  labels=['第0类', '第1类', '第2类'],  shadow=True,  autopct='%.1f%%',  pctdistance=0.7  )  plt.title('各类会员对总销售额的贡献')  plt.show() |

**\*代码详见：D:\python\_files\百货商场数据分析\项目任务\code\task\_4\task4-1\_2.py\***

# 小结

本案例向读者展示了如何使用python对百货商场的销售数据进行处理，通过对销售数据的清洗、处理、提取关键特征、提取价值特征、构建新特征、特征建模等操作，对商场的会员进行了画像，并且总结会员特点给出相应的营销建议。