

大型百货商场会员画像描绘

摘 要

• 本文针对在零售行业里，当前电商的发展使商场会员不断流失，给零售运营商带来了严重损失，给线下零售运营商带来了巨大抨击。因此运营商需要有针对性地实施营销策略来加强与会员的良好关系，所以需要通过对会员价值体现的数据处理得出的一系列有效途径，通过对统计分析会员的各项信息，得到一系列能够直观体现的图表，实现了对会员与大中的精细化管理，以及需求关系，更好的进行产品销售和服务。

首先对附件数据进行预处理，对附件数据进行有效筛选。

对于问题一，主要分为三部分，一是分析该商场会员的消费特征，分别通过性别和年龄分段两个基本属性，对人数占比、总消费金额占比、高消费人群占比进行分类探讨，同时分析男性会员和女性会员对不同商品的购买次数进行词云统计搜索，从而得到在该商场女性会员消费占主体，且高消费需求多，并且年龄差异对于该商场消费也是重要因素，与此同时一二季度的消费金额与次数明显高于三四季度，呈现下降趋势，且该商场在 2015 年-2018 年总体消费数据呈现上升趋势，再者男女会对于消费类别都倾向于购买服饰品。二是比较会员与非会员群体的差异，通过比较会员人群与非会员人群的平均消费金额、最高消费金额与最低消费金额之差以及消费金额方差，通过比较得出会员客户总消费金额、平均消费金额都大于非会员客户，但是会员中消费金额值差异比非会员人群较大，

三是说明会员群体给商场带来的价值，通过比较会员与非会员消费的差异以及会员总体的各项消费数据，如消费金额总数、消费总次数、单次消费金额，得出结论，一是会员通常具有较高的消费能力，二是会员通常对商场更具忠诚度，和更高的复购率，三是会员的客单价（单次消费金额）通常较高，更倾向高消费。

对于问题二，针对会员的消费情况建立能够刻画每一位会员购买力的数学模型，以便能够对每个会员的价值进行识别，我们利用 RFM 模型以及 K-means 聚类算法处理会员客户的相关数据，设定了相关的价值等级评定基础属性，计算每个会员消费数据，然后作价值组别分类，得到

关键词：消费金额；消费频率；购买力；RFM 模型

一、 问题重述

在零售行业中，会员价值体现在持续不断地为零售运营商带来稳定的销售额和利润，同时也为零售运营商策略的制定提供数据支持。零售行业会采取各种不同方法来吸引更多的人成为会员，并且尽可能提高会员的忠诚度。当前电商的发展使商场会员不断流失，给零售运营商带来了严重损失。此时，运营商需要有针对性地实施营销策略来加强与会员的良好关系。比如，商家针对会员采取一系列的促销活动，以此来维系会员的忠诚度。有人认为对老会员的维系成本太高，事实上，发展新会员的资金投入远比采取一定措施来维系现有会员要高。完善会员画像描绘，加强对现有会员的精细化管理，定期向其推送产品和服务，与会员建立稳定的关系是实体零售行业得以更好发展的有效途径。

附件中的数据给出了某大型百货商场会员的相关信息：附件 1 是会员信息数据；附件 2 是近几年的销售流水表；附件 3 是会员消费明细表；附件 4 是商品信息表，一般来说，商品价格越高，盈利越高；附件 5 是数据字典。请建立数学模型解决以下问题：

(1) 分析该商场会员的消费特征，比较会员与非会员群体的差异，并说明会员群体给商场带来的价值。

(2) 针对会员的消费情况建立能够刻画每一位会员购买力的数学模型，以便能够对每个会员的价值进行识别。

二、 问题分析

2.1 问题一分析

2.1.1 针对该商场会员客户群体的消费特征分析；

- 以附件 1 中所提供的 kh（会员卡号）为标识，在附件 3 中筛选出在 2015 年 1 月到 2017 年 12 月之间的消费明细，仅对处在这个区段的数据进行量化，总结消费特征；
- 将筛选过后的数据按照本地会员性别以及本地会员的年龄段划分成四组基础对象进行数据分析
- 将这四组基础对象分别进行年度总消费金额、年度总消费次数、年平均消费金额、年平均消费频率、以及每年各季度消费频率和金额这六项数据，进行数据分析并作为该商场会员客户群体的消费特征
- 将商品金额按照降序排序，取 sj 大于等于 10000 的商品作为衡量标准，在本地会员消费表中筛查购买了 10000 元以上商品的本地客户，再按照男女、年龄分段分析会员客户群体的消费特征；

2.1.2 针对该商场会员客户与非会员客户群体在消费特征上差异的分析；

- 以附件 2 中的时间周期（2016/1/—2017/9）为标准，结合附件 3 中会员客户的消费数据，筛选出会员与非会员的消费流水表；
- 对附件 2 与附件 3 的数据按 djh 作为特征标识进行筛选后，得到的会员消费表和非会员消费表作为数据分析的基础对象；
- 通过两种客户间的、总消费金额、平均购买金额、单次消费最高金额与单次消费最低金额之差、消费金额方差以及标准差来进行差异

的分析；

d) 通过筛查会员消费表和非会员消费表单次消费金额超过 10000 元的客户，进行对比差异分析；

2.1.3 针对该商场会员客户群体给商场带来的价值分析

a) 通过得到的该商场的会员客户群体的消费特征，进行数据分析，得出各项消费特征对与价值影响趋势

b) 通过该商场会员客户与非会员客户群体在消费特征上的差异分析后，研究会员相比非会员带来的价值差异，总结会员客户群体给商场带来的价值并分析

2.2 问题二分析

针对会员的消费情况建立能够刻画每一位会员购买力的数学模型，以便能够对每个会员的价值进行识别。

2.2.1 购买力理论知识如下

购买力含义：

购买力是指个人、家庭或组织在一定时间内能够购买商品或服务的经济实力。它反映了一个人或团体能够以其可支配的收入或资源购买多少物品或服务的能力。

影响购买力的因素：

购买力受到多个因素的影响，包括个人的收入水平、财务状况，以及物价水平和通货膨胀率等。较高的购买力意味着一个人或团体能够购买更多数量、更高价值的商品或服务。

购买力的作用：

购买力在市场经济中具有重要的作用。对于企业和商家来说，了解消费者的购买力可以帮助他们确定产品的价格和定位策略，以满足不同层次和需求的消费者。对于经济政策制定者和市场调研人员来说，购买力数据可以提供有关经济发展、消费趋势和市场规模的信息，从而指导政策和决策的制定。

需要注意的是，购买力是一个相对概念，通常是与特定时间段、地区或比较基准相关联。购买力的测量可以使用各种指标和方法，包括国内生产总值（GDP）和消费者物价指数（CPI）等。

2.2.2 分析处理：

- (1) 设定该商场会员的所有人的收入水平与消费次数、消费金额存在正相关关系，且物价水平变化值在一定范围内
- (2) 利用 RFM 模型以及 K-means 聚类算法处理该商场的会员消费数据
- (3) 将最近购买时间(Recency)、购买频率(Frequency)和购买金额(Montary)设为评判购买力的基础属性，并设置相应价值等级评定
- (4) 对处理好的数据进行组别分类
- (5) 总结数据处理结果

三、 基本假设与符号说明

3.1 基本假设:

在本研究中，我们基于以下假设进行研究：

数据可靠性假设：

- 1、假设收集到的数据是准确、完整和可靠的，并且遵循了相关的数据收集准则和标准。
- 2、假设所获取的 2015 年、2016 年、2017 年全年都有购买记录，即使用其余年份的季度平均数据可以填充缺失年份的缺失值。
- 3、假设 2015 年至 2017 年的会员消费情况能够反应本次研究探讨分析的问题，即 2018 年的会员消费数据对于结果不产生影响。
- 4、假设 1958 年之前的数据遵守于 1958 年至 2018 年的消费规律，即 1958 年之前的消费记录不会对本次研究产生影响。
- 5、假设商品折扣及商品退换货，所带来的数据误差忽略不计。
- 6、假设一个会员客户有且仅有一个会员卡号，即会员卡号是会员客户的唯一标识。
- 7、假设剔除小部分数据之后不会影响整体的数据结果。
- 8、假设采取抽样调查的数据处理方法，能够体现该商场会员客户的一些消费特征

随机样本假设：

- 1、假设采用的样本是随机选择的，并且能够代表目标人群的特征和分布。
- 2、假设采用单次消费金额超过 10000 可以作为最高消费类别。
- 3、假设采用单次消费金额超过 10000 视作高消费人群。
- 4、假设所有的客户数据都是真实存在且每个客户样本都无特殊需求。

独立观测假设：

- 1、假设每个观测之间是相互独立的，即每个观测值不受其他观测值的影响。
- 2、假设每年的社会环境都是安稳相同的，无社会动荡等影响条件
- 3、假设该商场内的销售人员都有合格的导购能力，即不对客户购买产生影响
- 4、设定该商场会员的所有人的收入水平与消费次数、消费金额存在正相关关系，且物价水平变化值在一定范围内

限制范围假设：

我们承认本研究具有一定的限制范围，包括样本规模、研究时间、地理位置等。这些限制将对结果的泛化性和推广性产生一定的影响。

请注意，以上假设是在我们研究的特定背景下建立的，可能并不适用于其他情况或领域。

3.2 符号说明:

符号	含义
kh	会员卡号
je	消费金额
xb	性别

sl	销售数量
spbm	商品编码
csny	出生年月
djh	单据号
sj	售价
djsj	登记时间
dtime	会员消费产生时间
LIQUID	数据可视化-水滴图
$\frac{1}{n} \sum_{i=1}^n (X_i - \bar{X})^2$	方差
$\sqrt{\frac{1}{n} \sum_{i=1}^n (X_i - \bar{X})^2}$	标准差
spmc	商品名称

四、 数据预处理

附件中的数据给出了某大型百货商场会员的相关信息：附件 1 是会员信息数据；附件 2 是近几年的销售流水表；附件 3 是会员消费明细表；附件 4 是商品信息表，一般来说，商品价格越高，盈利就越高；附件 5 是数据字典。

结合商场运营的实际消费情况，对附件中的数据进行以下合理化处理：

(1) 以 csny 为评判标准同时以 2018 年为参考年限，附件中年龄周岁小于 20 岁和大于 60 岁的用户数据视作无效数据，筛选并删除该数据；

(2) 在附件 1 中，对其中的数据作异常值处理，设 csny、xb、djsj 不明确为空值的视作无效数据，筛选并删除该数据；

(3) 附件 2 为销售流水表，将其中数据按 djh 为特征标识与附件 3 会员消费明细表，进行对比筛选，分出会员与非会员的消费情况表；

(4) 剔除附件 3 中消费金额为负值和消费数量为负值的数据；

(5) 附件 3 为消费明细表，将其中数据按 kh 为特征标识与附件 1 会员信息表，进行对比筛选，分出本地会员与非本地会员的消费情况表；

(6) 将会员消费表中 2016 年和 2017 年的平均数据填充 2015 年里的缺失值。

五、 问题一的模型建立与求解

5.1 针对分析该商场会员的消费特征：

(1) 年龄分段以及性别在本地会员的占比分析

模型建立：

确定目标：明确问题的目标是年龄分段和性别在所分析到的本地会员总人数中的占比，需要分析筛查出附件 3 会员消费表中含有的本地会员消费的

数据，然后从中计算出本地会员总人数的解答或结果。

收集数据：收集附件 1 和附件 3 的会员卡号以及性别，然后进行筛查和分析建模。

变量选择：本问题包含的都是从附件中筛查并提取的常量。

建立数学模型：按照百分比的模式计算占比

参数估计：对于本问题中的本地会员卡号采取抽样计算方法，保留小数点后两位。

模型求解：

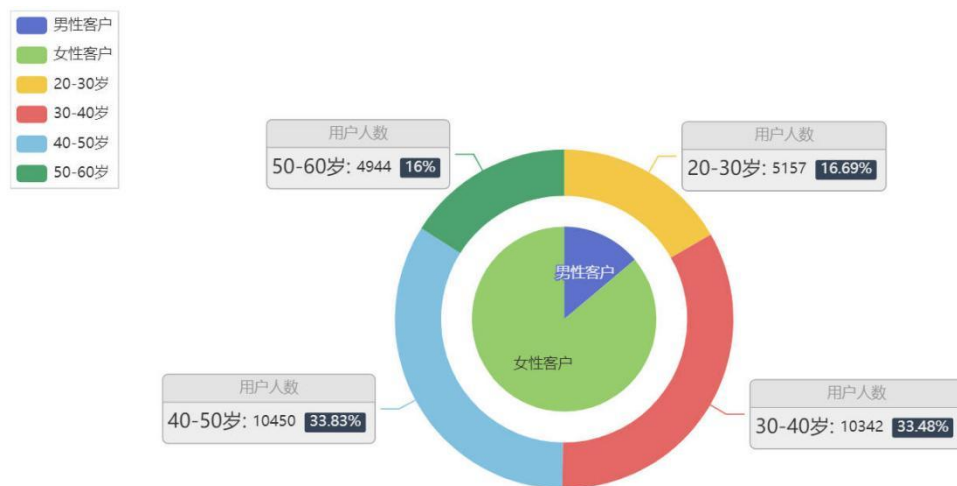
求解方法：1、各年龄段的会员卡号总数除以该商场中含有的所有的会员卡号总数。2、分别求出男性女性的会员卡号总数除以该商场中包含的所有的会员卡号总数。

程序实现：

- 1、对于附件 1 和附件 3 中的以 hykh 为特征标识进行筛选
 - a、先引入 pandas 包，以便需要 pandas 包相应的方法
 - b、之后将附件一和附件三表中数据通过 `pd.read_excel()` 方法读取其两个表
 - c、使用 `isin` 方法检查匹配项，从而在附件三表中得到附件一表中正式会员的索引
 - d、利用 `.loc[索引]` 方法得到每个索引行的数据并正整合起来得到一个基于附件一的会员消费明细表
 - e、最后利用 `.to_excel()` 方法打印出一张符合筛选要求的表
- 2、对于占比数据进行画图
 - a、将所提供附件数据表处理好后进行分类，整合分类
 - b、利用 python 读取 Excel 表格中的数据，再结合 pycharm 中对数据可视化所提供的 `pyecharts` 包，生成不同的数据表格
 - c、利用 `pyecharts.options` 和 `pyecharts.charts` 函数对会员消费数据表格进行处理，生成可视图表

参数调整：对模型中的参数进行调整，以获得更准确和可靠的结果。

求解过程：运行程序，使用所选的求解方法对模型进行求解。这可能涉及迭代、优化算法的收敛等。



结果分析:

该商场的主要消费人群是女性会员客户，且主要是年龄处于 30 至 40 岁和 40 至 50 岁区间的女性消费人群。

(2) 年龄分段以及性别对于 2015 年至 2017 年的总消费金额的占比分析

模型建立:

确定目标: 明确问题的目标是年龄分段和性别在所分析到的本地会员总消费金额中的占比, 需要分析筛查出附件 3 会员消费表中含有的本地会员消费的数据, 然后从中计算出本地会员总消费金额的解答或结果。

收集数据: 收集附件 1 和附件 3 的 kh、xb 以及 je, 然后进行筛查和分析 and 建模。

变量选择: 本问题包含的都是从附件中筛查并提取的常量。

建立数学模型: 按照百分比的模式计算占比

参数估计: 对于本问题中的本地会员卡号以及消费金额采取抽样计算方法, 保留小数点后两位。

模型求解:

求解方法: 1、各年龄段的会员消费金额总数除以该商场中含有的所有的会员消费金额总数。2、分别求出男性女性的会员消费金额总数除以该商场中包含的所有的会员消费金额总数。

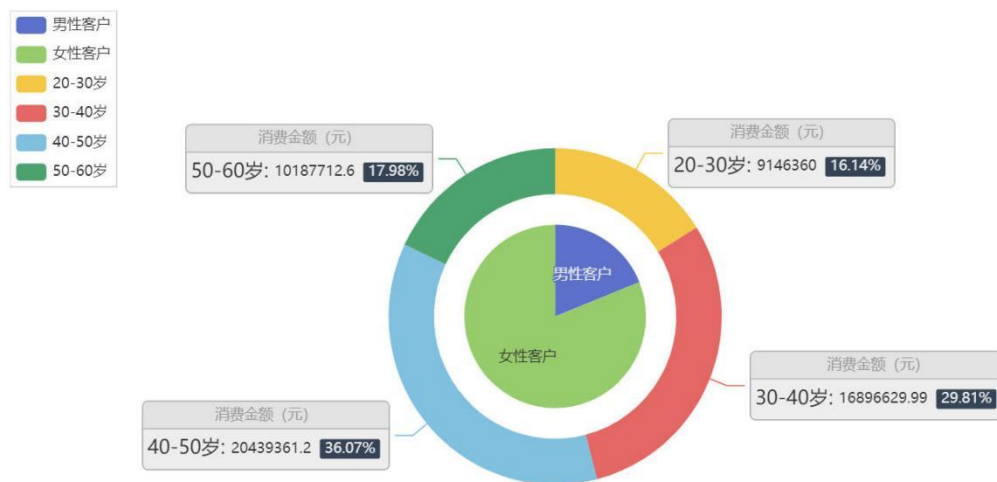
程序实现:

1、筛选方式处理同问题年龄分段以及性别在本地会员的占比分析

2、python 数据可视化实现图表

参数调整: 对模型中的参数进行调整, 以获得更准确和可靠的结果。

求解过程: 运行程序, 使用所选的求解方法对模型进行求解。这可能涉及迭代、优化算法的收敛等。



结果分析：

该商场女性会员客户所带来的价值远大于男性会员所带来的价值，且是年龄范围在 40 至 50 岁的女性会员客户。

(3) 2015 - 2017 不同季度会员用户消费次数

模型建立：

确定目标：明确问题的目标是 2015 - 2017 年区间内不同季度会员的单季度会员消费次数，需要找到 2015 - 2017 年这三年内，单年内不同季度的消费趋势以及这三年内会员消费次数的总体趋势。

收集数据：收集附件 1 和附件 3 内 kh、dtime 整理并筛查，然后进行筛选统计和分析

建立数学模型：根据整数求和的方式，分别计算 2015 - 2017 这三年不同季度的会员客户消费次数

参数估计：将 dtime 的次数，粗略估计成会员客户消费次数，即产生一次 dtime，便将其视为一次会员客户消费

模型求解：

求解方法：将筛查好的 hykh 以及 dtime 进行分季度统计，并绘制折线图以直观表现消费次数趋势

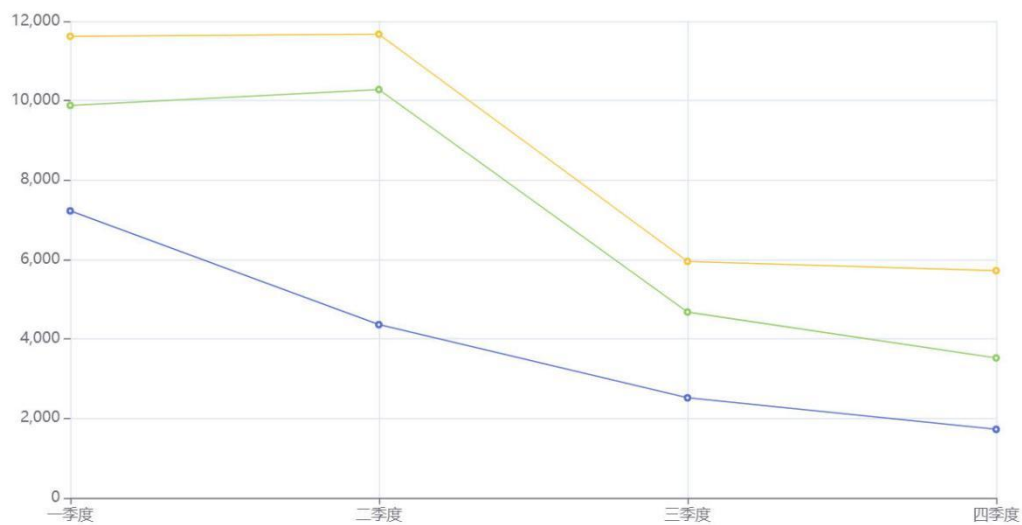
程序实现：

- 1、筛选方式处理同问题年龄分段以及性别在本地会员的占比分析问题
- 2、python 数据可视化处理

参数调整：对模型中 2015 年所缺失的 9、10、11 月份的 dtime 以及 2016 年所缺失的 2 月份的 dtime 参数进行调整，将含有缺失月份的年份的其余两年的同月份进行统计求平均值，用该平均值填充所缺失的数值，以获得更准确和可靠的结果。

求解过程：运行程序，使用所选的求解方法对模型进行求解。

2015-2017不同季度会员用户消费次数表



结果分析：

该商场 2015 年至 2017 年的年度总会员消费次数是呈现上升趋势的，且一二季度会员用户消费次数大于三四季度，呈现下降趋势。在 2015 年之后每年第二季度会员消费频率达到本年的消费峰值，在该季度会员消费次数频繁。

(4) 2015 - 2017 不同季度会员用户消费总额表

模型建立

确定目标：明确问题的目标是 2015 - 2017 年区间内不同季度会员的单季度会员消费总额，需要找到 2015 - 2017 年这三年内，单年内不同季度的消费趋势以及这三年内会员消费的总体趋势。

收集数据：收集附件 1 和附件 3 内 kh、je 整理并筛查，然后进行筛选统计和分析

建立数学模型：根据整数求和的方式，分别计算 2015 - 2017 这三年不同季度的会员客户消费金额

参数估计：仅收集 je 为正的数值，剔除 je 为 0 或负的数值，进行统计

模型求解：

求解方法：将筛查好的 hykh 以及 je 进行分季度统计，并绘制折线图以直观表现会员客户消费趋势

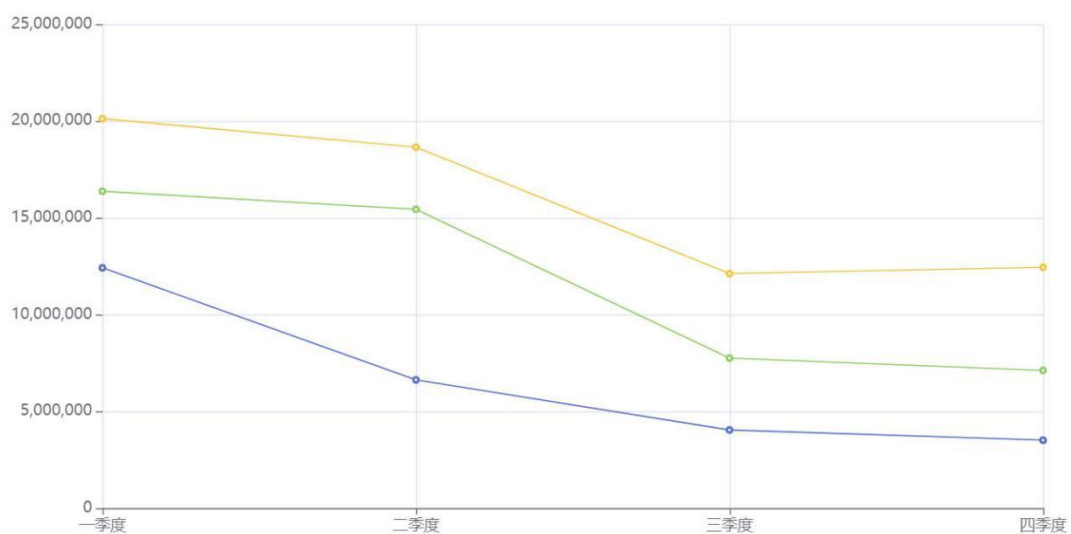
程序实现：

- 1、筛查方式处理同问题年龄分段以及性别在本地会员的占比分析问题。
- 2、python 数据可视化处理

参数调整：对模型中 2015 年所缺失的 9、10、11 月份的 je 数值以及 2016 年所缺失的 2 月份的 je 数值参数进行调整，将含有缺失月份的年份的其余两年的同月份进行统计求平均值，用该平均值填充所缺失的数值，以获得更准确和可靠的结果。

求解过程：运行程序，使用所选的求解方法对模型进行求解。

2015-2017不同季度会员用户消费总额表



结果分析：

该商场 2015 年至 2017 年的年度总会员消费总金额是呈现上升趋势的，且一、二季度会员用户带来的价值大于三、四季度，呈现下降趋势。在 2015 年之后每年第二季度会员消费总金额达到本年的消费峰值，即在该季度商场所能营收的金额最高。

(5) 对于高消费人群分别在男女人数中的占比以及在总人数中的占比

模型建立：

确定目标：明确问题的目标是高消费人群中总共含有多少人，其分别在男女人数中的占比以及在总人数中的占比，得到高消费人群于非高消费人群的比例。

收集数据：将附件 1 和附件 3 内的 kh、xb、je 整理并筛查，然后分别统计男性会员客户，女性会员客户，总会员客户人数以及单次消费金额超过 10000 的会员卡号，并在其中筛选男女会员客户。

变量选择：本问题研究选取的数据都来源于附件。

建立数学模型：分别统计各项数据总数并作百分比计算求得高消费人群分别在男女人数中的占比以及在总人数中的占比。

参数估计：将会员卡号在单次消费中超过 10000 的消费金额设置为高消费，并将含有高消费的会员客户认定为本问题研究的高消费人群，并将计算所得的数据保留小数点后两位

模型求解：

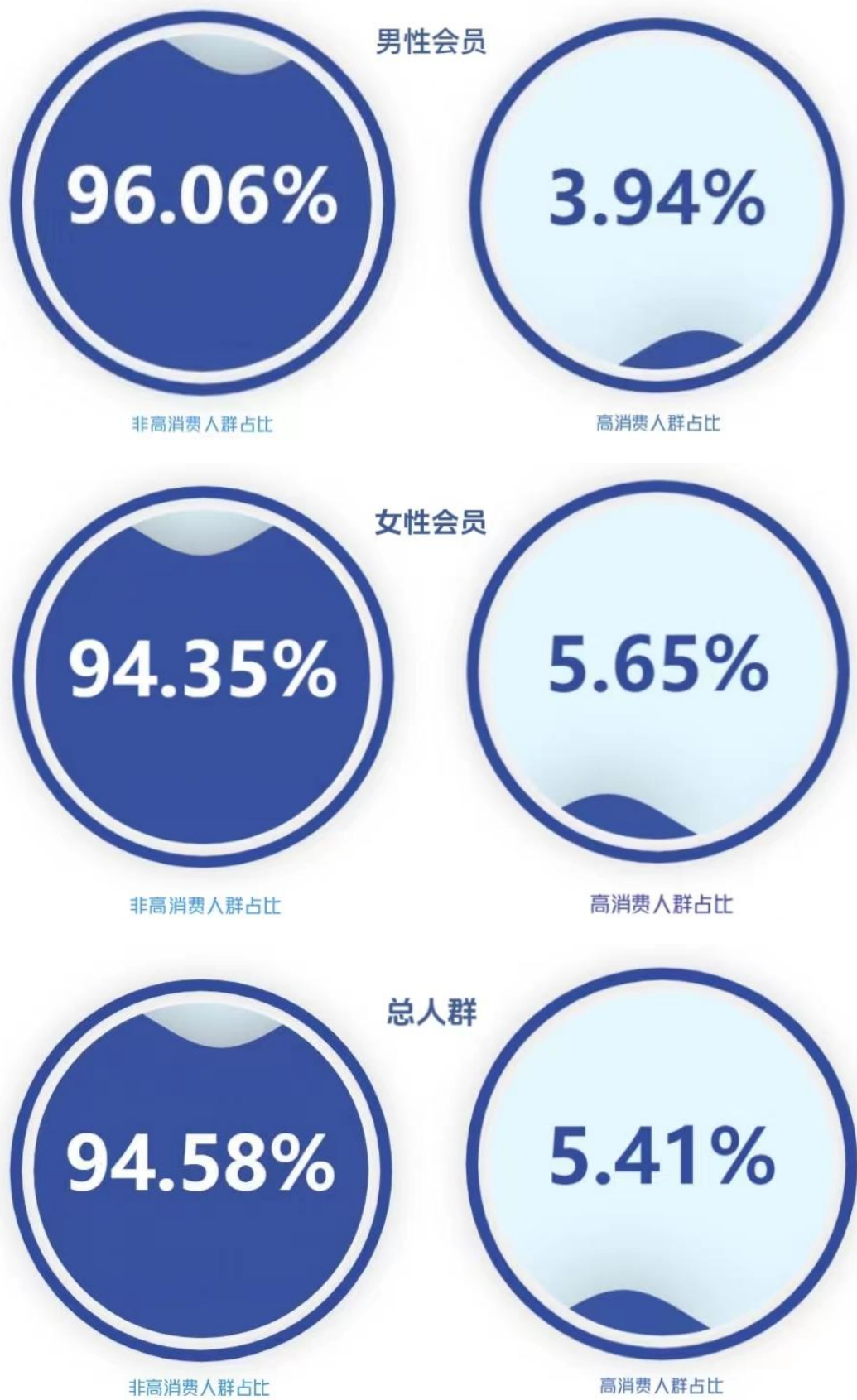
求解方法：将得到的高消费人群分别在男女人数中的占比以及在总人数中的占比，使用水球图 Liquid 呈现占比大小，从而反应高消费人群对于该商场的价值

程序实现：：

- 1、筛查方式处理同问题年龄分段以及性别在本地会员的占比分析问题。
- 2、python 数据可视化处理

参数调整：将无法识别的 je、xb、kh 数值以及 je、xb、kh 为负值或 0 的数

值剔除，。
求解过程：运行程序，使用所选的求解方法对模型进行求解。



结果分析：

该商场高消费人群占比极小，且女性会员对于高质量要求大于男性会员。

(6) 对于男女大众的消费倾向的词云搜索

模型建立

确定目标：明确问题的目标是筛查会员客户对于不同商品的消费次数，通过词云搜索，体现该商场会员客户的购买倾向，需要找到该商场客户会员对于不同商品的消费次数。

收集数据：收集附件 1 和附件 3 内的 kh、xb、spmc 数据进行筛查，得到关于该商场本地会员对于不同商品的消费次数

变量选择：随机抽样 3 组数据进行对比

建立数学模型：分别对男女不同的对不同商品的消费次数，进行统计整理，然后通过词云搜索凸显一些购买次数较高的商品名称

参数估计：采用随机抽样的调查方法，通过多组实验数据来体现购买商品名称

模型求解:

求解方法：通过统计男女会员客户不同的对于不同商品的消费次数，然后整合数据，经过词云搜索，得到一份能够体现不同购买次数的词云图，分析词云图中关键词的不同出现次数，体现该商场男女会员客户不同消费倾向。

程序实现:

- 1、筛查方式处理同问题年龄分段以及性别在本地会员的占比分析问题。
- 2、python 数据可视化处理

参数调整：剔除数据异常的数据，设置不同的抽样调查的样本

求解过程：运行程序，使用所选的求解方法对模型进行求解。





结果分析：

该商场的男性和女性会员消费倾向都是服饰品

5.2 针对比较会员与非会员群体的差异：

模型建立：

确定目标：明确问题的目标是通过对比会员与非会员的各项基础数据进行比较，找到相关差异，需要找到对于消费会员在哪些方面优于非会员，在哪些方面又还有不足。

收集数据：收集附件 2 与附件 3 还有附件 1 中的 kh、djh、je 数据进行整理与筛查，然后分别进行会员与非会员的消费金额数据进行统计。

变量选择：随机抽取 5000 份数据进行统计比较

建立数学模型：分别对会员与非会员的 5000 份消费金额数据进行统计，利用平均值、方差公式、标准差公式求出相应数据

模型求解：

求解方法：采用随机抽样调查的方法，抽取筛选后的表格中 5000 份会员与非会员的消费金额数据，然后进行统计，再利用平均值公式、方差公式、标准差公式对数据进行处理得到，有关会员与非会员消费金额对比图，通过图表中呈现的数据进行对比得出结果

客户	消费总人数	总消费额	平均消费额	最高和最低消费的差值	标准差	方差
会员客户	5000	7816350	1584.82	63500	3161.77	9996791
非会员客户	5000	6868808	1378.45	50400	2920.85	8531369

结果分析：

该商场会员客户带来的价值是大于非会员用户的，且会员用户对于高质量消

费的需求大于非会员用户。

5.3 针对会员群体给商场带来的价值

模型建立：

确定目标：明确问题的目标是通过对该商场的会员消费特征进行统计分析，找出相关关系，需要找到对于消费会员在哪些方面存在价值体现

收集数据：利用问题“针对分析该商场会员的消费特征”中含有的相关消费特征的数据进行整理统计

求解方法：通过处理各项会员消费特征的数据进行数据归纳统计，并绘制相关表格，通过比较各表中的数据得出消费会员在哪些方面存在价值体现

性别	高消费人数	非高消费人数	总人数
男	169	4122	4291
女	501	26101	26602

2015 年	消费总金额（元）	消费总次数
一季度	12412000	7224
二季度	6620033	4360
三季度	4033125	2518
四季度	3504325	1724

2016 年	消费总金额（元）	消费总次数
一季度	3955730	2655
二季度	8813845	5918
三季度	3711825	2157
四季度	3605935	1797

2017 年	消费总金额（元）	消费总次数
一季度	3751484	1739
二季度	3211472	1393
三季度	4266233	1276
四季度	5329711	2198

5.4 问题一的结果分析

会员群体给商场带来的价值表现在以下几个方面，从会员与非会员的消费特征可以更深入地了解这些价值：

1、高消费能力：会员通常具有较高的消费能力，他们愿意花费更多的金钱购买商品或享受服务。相比之下，非会员的消费金额相对较低。

2、忠诚度和复购率：会员通常对商场更具忠诚度，更有可能经常光顾商场并进行复购。非会员则往往没有这种忠诚度，容易受到竞争对手的吸引而流失。

3、高客单价：会员的客单价（单次消费金额）通常较高，因为他们通常倾向于购买更昂贵的商品或服务。非会员的客单价相对较低。

4、促进口碑和营销效果：会员作为商场的忠实顾客，对商场的口碑和品牌形象具有积极影响。他们更有可能推荐商场给其他人，并参与商场的营销活动。

5、数据分析和个性化营销：会员消费数据的收集和分析能够帮助商场深入了解消费者行为和偏好，并进行个性化的营销和推荐。非会员则无法提供这种数据支持。

通过比较会员和非会员的消费特征，商场可以更好地了解会员对于商场的价值和推动力。同时，商场也可以通过提供更好的会员权益和服务来吸引更多的顾客成为会员，进一步提升商场的价值。

六、 问题二的模型建立与求解

6.1 问题二的模型建立

RFM 模型是一种用于客户分析和市场细分的有效工具，采用 RFM 模型，通过分析会员客户最近购买时间（Recency）、购买频率（Frequency）和购买金额（Monetary）等基础属性，对会员客户进行分类和评估。分析会员客户所被划分的类别以及模型评估来进行价值评定，得出每一位会员对于商场的价值。

K-means 聚类算法是一种常用的划分聚类算法，适用于将数据点划分为不同的簇，本题运用 K-means 聚类算法处理根据计算得到的 R、F、M 值，对会员客户对于商场的价值进行分类。

6.1.1 设置基础属性：

（1）最近购买时间 Recency：衡量会员客户最近一次购买的时间间隔，较短的时间间隔表示客户更有可能继续购买，客户活跃度高，可能是高价值会员客户。

（2）购买频率 Frequency：衡量会员客户在一段时间的消费次数，消费次数越多，说明会员客户在商场消费次数越频繁。

（3）购买金额 Monetary：衡量客户在一段时间内的消费总金额，较高的购买总金额表示该会员客户对于本商场的价值更高，高消费金额的会员客户往往是高价值会员客户，对商场的收入贡献较大。

6.1.2 设置价值组别：

（a）高价值会员客户：最近购买时间较短，消费频次较高，消费金额较高。

（b）潜力会员客户：最近购买时间较短，消费频次较低，消费金额较低。

（c）忠诚会员客户：最近购买时间较长，消费频次较高，消费金额较高。

（d）流失会员客户：最近购买时间较长，消费频次低，消费金额较低

6.2 问题二的模型求解：

6.2.1 RFM 模型

数据准备：收集附件 1 与附件 3 中的会员客户的消费数据，包括 kh、dtime、

je 等信息。将这些数据组织成适合分析的数据集，确保数据的准确性和完整性。

数据处理：根据该商场的会员客户的消费数据，计算每个客户的 R、F、M 值。将最近一次消费产生时间与当前日期相减，得到 R 值；计算总消费次数得到 F 值；计算总消费金额得到 M 值。

分类：使用 K-means 聚类算法处理得到的 RFM 数值

数据总结：对于不同的会员客户价值组别，进行进一步的分析和解释

6.2.2 K-means 聚类算法

1. 初始化：选择要划分的簇的个数 K，并随机初始化 K 个簇的中心点。
2. 计算距离：对于每个数据点，计算其与各个簇中心点之间的距离。常用的距离度量方法是欧氏距离或曼哈顿距离。
3. 分配簇：将每个数据点划分到距离最近的簇中，即将数据点分配给与其最近的簇。
4. 更新簇中心：对于每个簇，重新计算其数据点的平均值，作为新的簇中心。

重复步骤 2 和 4，直到达到收敛条件。收敛条件可以是簇中心不再发生变化，或者达到预设的迭代次数。

得到最终的聚类结果，每个数据点都被划分到了一个簇中。

6.3 问题二的结果结果分析

七、 模型评价与改进方向

八、 参考文献

九、 附录

9.1 数据筛查代码

筛查本地会员信息和消费明细代码

```
import pandas as pd
df = pd.read_excel('D:\桌面\案例练习 1-计算机-数据分析类\附件 1-会员信息表.xlsx')
df1 = pd.read_excel('D:\桌面\案例练习 1-计算机-数据分析类\附件 3-会员消费明细表.xlsx')
index_list = df1[df1['会员卡号'].isin(df['会员卡号'])].index
df2 = df1.loc[index_list]
index_list2 = df2.drop_duplicates(subset=['会员卡号']).index
df3 = df2.loc[index_list2]
merge_df = df3.merge(df, on='会员卡号', how='left')
merge_df['会员消费产生的时间'] = merge_df['会员消费产生的时间'].dt.date
```



```
merge_df.to_excel('附件 7-会员信息消费明细表.xlsx', index=False)
```

筛查本地会员和非本地会员

```
import pandas as pd
df = pd.read_excel('D:\桌面\案例练习 1-计算机-数据分析类\附件 2-销售流水
表.xlsx')
df1 = pd.read_excel('D:\桌面\作业 1\附件 7-会员信息消费明细表.xlsx')
index_list = df[~df['单据号'].isin(df1['单据号'])].index
df2 = df.loc[index_list]
df2.to_excel('A.xlsx', index=False)
```

9.2 pycharm 数据可视化代码

饼状图代码

```
# coding=UTF-8
import pycharts.options as opts
from pycharts.charts import Pie

inner_x_data = ["男性客户", "女性客户"]
inner_y_data = [4291, 26602]
inner_data_pair = [list(z) for z in zip(inner_x_data, inner_y_data)]

outer_x_data = ["20-30 岁", "30-40 岁", "40-50 岁", "50-60 岁"]
outer_y_data = [5157, 10342, 10450, 4944]
outer_data_pair = [list(z) for z in zip(outer_x_data, outer_y_data)]

(
    Pie()
    .add(
        series_name="用户人数",
        data_pair=inner_data_pair,
        radius=[0, "30%"],
        label_opts=opts.LabelOpts(position="inner"),
    )
    .add(
        series_name="用户人数",
        radius=["40%", "55%"],
        data_pair=outer_data_pair,
        label_opts=opts.LabelOpts(
            position="outside",
            formatter="{a|{a}}{abg|}\n{hr|}\n{b|{b}:}{c} {per|{d}%"
",
        background_color="#eee",
```

```

        border_color="#aaa",
        border_width=1,
        border_radius=4,
        rich={
            "a": {"color": "#999", "lineHeight": 22, "align":
"center"},
            "abg": {
                "backgroundColor": "#e3e3e3",
                "width": "100%",
                "align": "right",
                "height": 22,
                "borderRadius": [4, 4, 0, 0],
            },
            "hr": {
                "borderColor": "#aaa",
                "width": "100%",
                "borderWidth": 0.5,
                "height": 0,
            },
            "b": {"fontSize": 16, "lineHeight": 33},
            "per": {
                "color": "#eee",
                "backgroundColor": "#334455",
                "padding": [2, 4],
                "borderRadius": 2,
            },
        },
    ),
)
    .set_global_opts(legend_opts=opts.LegendOpts(pos_left="left",
orient="vertical"))
    .set_series_opts(
        tooltip_opts=opts.TooltipOpts(
            trigger="item", formatter="{a} <br/>{b}: {c} ({d}%)"
        )
    )
    .render("001.html")
)

```

```

# coding=UTF-8

```

```

import pyecharts.options as opts
from pyecharts.charts import Pie

```

```

inner_x_data = ["男性客户", "女性客户"]

```

```

inner_y_data = [10707489.33, 45962607.1]
inner_data_pair = [list(z) for z in zip(inner_x_data, inner_y_data)]

outer_x_data = ["20-30 岁", "30-40 岁", "40-50 岁", "50-60 岁"]
outer_y_data = [9146360, 16896629.99, 20439361.2, 10187712.6]
outer_data_pair = [list(z) for z in zip(outer_x_data, outer_y_data)]

(
    Pie()
    .add(
        series_name="消费金额 (元) ",
        data_pair=inner_data_pair,
        radius=[0, "30%"],
        label_opts=opts.LabelOpts(position="inner"),
    )
    .add(
        series_name="消费金额 (元) ",
        radius=["40%", "55%"],
        data_pair=outer_data_pair,
        label_opts=opts.LabelOpts(
            position="outside",
            formatter="{a|{a}}{abg|}\n{hr|}\n{b|{b}:}{c} {per|{d}%} ",
            background_color="#eee",
            border_color="#aaa",
            border_width=1,
            border_radius=4,
            rich={
                "a": {"color": "#999", "lineHeight": 22, "align": "center"},
                "abg": {
                    "backgroundColor": "#e3e3e3",
                    "width": "100%",
                    "align": "right",
                    "height": 22,
                    "borderRadius": [4, 4, 0, 0],
                },
                "hr": {
                    "borderColor": "#aaa",
                    "width": "100%",
                    "borderWidth": 0.5,
                    "height": 0,
                },
                "b": {"fontSize": 16, "lineHeight": 33},
                "per": {
                    "color": "#eee",

```

```

        "backgroundColor": "#334455",
        "padding": [2, 4],
        "borderRadius": 2,
    },
},
),
)
.set_global_opts(legend_opts=opts.LegendOpts(pos_left="left",
orient="vertical"))
.set_series_opts(
    tooltip_opts=opts.TooltipOpts(
        trigger="item", formatter="{a} <br/>{b}: {c} ({d}%)"
    )
)
.render("002.html")
)

```

折线图代码

```

# coding=UTF-8
import pyecharts.options as opts
from pyecharts.charts import Line

x_data = ["一季度", "二季度", "三季度", "四季度"]
y_data = [1000, 2000, 3000, 4000, 5000]

(
    Line()
    .add_xaxis(xaxis_data=x_data)
    .add_yaxis(
        series_name="2015 年",
        stack="总量",
        y_axis=[7224, 4360, 2518, 1724],
        label_opts=opts.LabelOpts(is_show=False),
    )
    .add_yaxis(
        series_name="2016 年",
        stack="总量",
        y_axis=[2655, 5918, 2157, 1797, ],
        label_opts=opts.LabelOpts(is_show=False),
    )
    .add_yaxis(
        series_name="2017 年",
        stack="总量",

```

```

        y_axis=[1739,1394,1276,2198,],
        label_opts=opts.LabelOpts(is_show=False),
    )

    .set_global_opts(
        title_opts=opts.TitleOpts(title="2015-2017 不同季度会员用户消费次数表"),
        tooltip_opts=opts.TooltipOpts(trigger="axis"),
        yaxis_opts=opts.AxisOpts(
            type_="value",
            axistick_opts=opts.AxisTickOpts(is_show=True),
            splitline_opts=opts.SplitLineOpts(is_show=True),
        ),
        xaxis_opts=opts.AxisOpts(type_="category", boundary_gap=False),
    )
    .render("003.html")
)

```

```

-----
# coding=UTF-8
import pyecharts.options as opts
from pyecharts.charts import Line

x_data = ["一季度", "二季度", "三季度", "四季度"]
y_data = [1000,2000,3000,4000,5000]

(
    Line()
    .add_xaxis(xaxis_data=x_data)
    .add_yaxis(
        series_name="2015 年",
        stack="总量",
        y_axis=[12412000,6620033,4033125,3504325,],
        label_opts=opts.LabelOpts(is_show=False),
    )
    .add_yaxis(
        series_name="2016 年",
        stack="总量",
        y_axis=[3955730,8813845,3711825,3605935,],
        label_opts=opts.LabelOpts(is_show=False),
    )
    .add_yaxis(

```

```

        series_name="2017 年",
        stack="总量",
        y_axis=[3751484, 3211472, 4366233, 5329711, ],
        label_opts=opts.LabelOpts(is_show=False),
    )

    .set_global_opts(
        title_opts=opts.TitleOpts(title="2015-2017 不同季度会员用户消费总额
表"),
        tooltip_opts=opts.TooltipOpts(trigger="axis"),
        yaxis_opts=opts.AxisOpts(
            type_="value",
            axistick_opts=opts.AxisTickOpts(is_show=True),
            splitline_opts=opts.SplitLineOpts(is_show=True),
        ),
        xaxis_opts=opts.AxisOpts(type_="category", boundary_gap=False),
    )
    .render("004.html")
)

```

水滴图代码

```

# coding=UTF-8
from pyecharts import options as opts
from pyecharts.charts import Grid, Liquid
from pyecharts.commons.utils import JsCode

l1 = (
    Liquid()
    .add("lq", [0.94350, 0.07], center=["65%", "50%"])
    .set_global_opts(title_opts=opts.TitleOpts(title="多个 Liquid 显示
"))
)

l2 = Liquid().add(
    "lq",
    [0.9435],
    center=["25%", "50%"],
    label_opts=opts.LabelOpts(
        font_size=50,
        formatter=JsCode(
            """function (param) {
                return (Math.floor(param.value * 10000) / 100) + '%';
            }"""
        ),
    ),
)

```

```

        position="inside",
    ),
)

grid      =      Grid().add(l1,      grid_opts=opts.GridOpts()).add(l2,
grid_opts=opts.GridOpts())
grid.render("005.html")

-----

# coding=UTF-8
from pyecharts import options as opts
from pyecharts.charts import Liquid
from pyecharts.commons.utils import JsCode

c = (
    Liquid()
    .add(
        "正常消费占比",
        [0.0541],
        label_opts=opts.LabelOpts(
            font_size=50,
            formatter=JsCode(
                """function (param) {
                    return (Math.floor(param.value * 10000) / 100) + '%';
                }"""
            ),
            position="inside",
        ),
    )
    .set_global_opts(title_opts=opts.TitleOpts(title="Liquid-数据精度"))
    .render("006.html")
)

-----

```

词云图代码

```

# coding=UTF-8
import pyecharts.options as opts
from pyecharts.charts import WordCloud

data = [
    #将 Excel 中的数据转换为元组的数据
]
(

```

```

WordCloud()
.add(series_name="热点分析", data_pair=data, word_size_range=[3, 46])
.set_global_opts(
    title_opts=opts.TitleOpts(
        title="热 点 分 析",
title_textstyle_opts=opts.TextStyleOpts(font_size=23)
    ),
    tooltip_opts=opts.TooltipOpts(is_show=True),
)
.render("007.html")
)

```

将 Excel 表格中的数据转换为 python 元组代码

```

# coding=UTF-8
import pandas as pd

# 读取 Excel 表格数据
df = pd.read_excel(r'C:\Users\30408\Desktop\数学建模\day01 数据集\词云数据.xlsx') # 替换为你的 Excel 文件路径

# 将数据转换为元组
data_tuples = [tuple(x) for x in df.values]

# 打印转换后的元组
for row in data_tuples:
    print(row, end=", ")
    print()

```