基于优化模型的蔬菜类商品自动定价与补货决策

摘　要

本文根据蔬菜类商品的销售历史数据，建立数学模型，以研究蔬菜类商品自动定价与补货决策问题。

针对问题一，我们首先对数据进行预处理，采用分类汇总、交叉分析等方法，初步整理了数据。基于预处理后的数据集，绘制不同品类或不同单品的直方图、箱线图等，对它们各自的销售量进行了描述性统计和相关分布规律分析。对于相互关系，利用 Pearson 相关系数进行相关性分析，分析不同品类间的相互关系；由于单品数较多，我们采用聚类分析，分别按月、季度进行 K-means 聚类，分析不同单品间的相互关系。

针对问题二，以品类为单位，我们通过线性回归拟合出各品类蔬菜销售量与成本加成定价的函数关系式。采用 ARIMA 时间序列预测出未来一周（2023 年 7 月 1-7 日）各品类的成本加成定价，为接下来的非线性目标优化提供数据。我们以日补货总量和定价为决策变量，考虑以品类为单位的平均损坏率和平均折扣率，将商超收益分为打折与未打折部分，设置其最大化为目标函数，再利用 SLSQP 算法求解出未来一周最优的日补货总量和定价策略。

针对问题三，由于销售空间有限，我们将 49 个筛选出的单品作为决策对象，构建混合整数非线性规划模型进行优化。利用基于 GRU 的 RNN 模型，预测可售单品在 7 月 1 日的成本，引入 0-1 变量表示该单品是否最终被选择为可售单品。以单品补货量和定价为决策变量，考虑最小陈列量、可售单品数量限制作为约束条件，在尽量满足市场需求的前提下，以商超收益最大化为优化目标。采用 BONMIN 求解器进行模型求解，根据 6 月 24-30 日可售品种的相关数据，给出 7 月 1 日的最优单品选择组合、补货量和定价策略。通过建模求解，可以在资源限制下平衡多目标，制定出商超收益最大化的补货方案。

针对问题四，我们认为可以提供有关市场、消费者和产品的更多信息。例如：竞争对手数据、市场需求数据、供应链数据、节假日涨价与供求情况、促销效果数据、天气数据、蔬菜变质速度等。这些数据将为商超经营提供更全面的信息，便于更准确地预测需求、制定补货计划、调整定价策略，助于提高市场竞争力。

本文对所建立的模型均进行了严格的模型检验，模型具有较高的合理性、准确性。所建立的混合整数非线性目标优划模型与实际紧密联系，可结合实际情况应用于求解生产进度问题、旅行推销员问题、工厂选址问题、背包问题及分配问题等，使得模型具有很好的通用性和推广性。

关键词：数据预处理 非线性 混合整数目标优化 蔬菜类商品

一、 问题重述

1.1 背景

随着人们生活水平的提高，对食品安全、品质和多样性的关注不断增加，越来越多的人选择在生鲜商超购买食材和生鲜产品。然而，蔬菜类商品的保鲜期较短且易受时间影响其品相。在生鲜商超中，如果当日蔬菜未能售出，通常无法再次销售。商超通常在凌晨 3 点至 4 点进行进货交易，但此时并不确切知道每个单品的库存情况和进货价格，这给商家在当天做出蔬菜品类的补货决策带来了一定的困难。因此，建立一个模型来自动定价和制定蔬菜类商品的补货决策对商超经营者来说非常重要，可以节省人力成本并增加盈利。

1.2 问题重述

本文的问题重述侧重于对题意的理解，不同的理解方式可能对后续建模方式产生较大的影响。

问题一：该问题的分析对象是不同品类和不同单品的蔬菜类商品，问题前半句以“或”划分了“不同品类”和“不同单品”，那么“不同品类”和“不同单品”可能需要分开分析。因此我们需要分析两方面内容，一是各自销售量的分布规律，二是不同品类之间及不同单品之间的相互关系。同时，需要注意单品中的编号信息，这表示不同的供应商来源，在分析单品时可能需要合并。该问题需要用到附件一和附件二的数据，其目的是洞察销量分布规律及品类和单品之间的内在联系，为后续的销量预测和优化决策奠定基础。

问题二：要求以品类为单位规划未来一周的补货计划。其中“日补货总量和定价策略”指明了决策变量的选择。题目首先要求我们分析各品类的销售总量与当前采用的成本加成定价之间的关系，这对后续求解决策变量至关重要。为了更贴合实际情况，需要对考虑损耗率和折扣率对规划求解的影响，因此本问需要综合附件一、附件二、附件三及附件四的数据，计算各蔬菜品类的销售总量，并与成本加成定价进行关联分析，以了解销售总量与定价之间的关系，最后以商超收益最大为目标，建立合理的优化函数。

问题三：该问题是对问题二的进一步分析，决策粒度发生变化，要求在单品层面制定补货计划，相对问题二的品类层面更加细致。问题三的决策变量是单品的补货量和定价策略，需要给出选中的单品在 7 月 1 日这一天的决策值。题目增加了更多的约束条件：可售单品总数限制在 27-33 个;每个单品最低订购量不少于 2.5 千克。属于硬性约束。同时，还需要考虑近期商品的市场需求，相当于对品类的总补货量约束了一个最小值。本题考虑使用整数规划的方法求解。

问题四：该问题的目的是为了更好地制定商超的蔬菜商品补货和定价决策，需

要关注的是与补货定价决策相关的新的数据，而非已有的数据，新的数据可以从商品、市场、环境等方面考虑。不限于已有的商品销售数据。同时要注意新的数据要与优化补货定价决策具有直接或间接的关系和作用。最后需要按照要求给出明确的意见和分析理由，不能仅提意见无理由分析。

二、 问题分析

2.1 问题一的分析

问题一：要求分析蔬菜各品类及单品销售量的分布规律及相互关系。

本题要求分析不同蔬菜品类和单品之间的销量分布规律及关联关系。首先，可以对历史销量数据进行统计分析，包括描述性统计分析和绘制直方图、盒须图等，观察不同品类和单品的销量分布情况，判断是否符合某种典型分布。其次，可以采用相关性分析方法，计算不同品类或不同单品之间的皮尔逊相关系数，判断它们之间在销量上是否存在显著的正相关或负相关。最后，还可以利用聚类分析方法，在不同时间粒度(如月度、季度)上对单品进行聚类，看不同单品之间是否存在相似的销量变化规律。以上分析可以全面洞察蔬菜类别和商品的销量分布特征以及关联关系，为后续决策提供依据。

2.2 问题二的分析

问题二：要求以品类为单位做补货计划，分析各蔬菜品类的销售总量与成本加成定价的关系。并给出 7 月 1 日补货总量和定价策略，从而使得商超收益最大。

本题要求以品类为单位制定未来一周的蔬菜补货和定价策略。首先，可以利用线性回归方法，建立各品类历史销量数据与定价或利润率之间的回归模型，分析两者的相关性。其次，可以采用时间序列分析方法，预测未来一周各品类的单位成本。然后，建立以补货总量和定价为决策变量，以周销售收益最大化为目标的非线性规划模型。最后，求解该规划模型，得到每类蔬菜在未来一周的优化补货总量和最优定价策略。这种方法综合考虑了数据特征、预测结果和收益最大化目标，可以使商超对每类蔬菜制定出更科学合理的补货和定价方案。

2.3 问题三的分析

问题三需要基于 2023 年 6 月 24 日至 6 月 30 日的可售品种数据，制定 7 月 1 日的单品层面补货策略。首先筛选出近一周内有销售的单品。然后根据题设要求，需要控制可售单品总数在 27-33 个之间，每个单品的订购量满足最小陈列量 2.5 千克。在满足这些硬性约束的前提下，补货计划还需兼顾不同品类的市场需求，以最大化商超收益为目标进行优化。

本文将通过建立混合整数规划模型求解此问题。引入 0-1 变量表示单品是否选择，

以单品订购量和定价为决策变量，加入相关约束条件。在满足各类约束的前提下，以单品层面收益的总和为目标函数，采用 BONMIN 求解器求出最优解，即可得到单品选择、订购量和定价决策，实现在资源约束条件下最大化商超收益。

相比直接凭经验制定方案，运筹学模型化可以更系统地描述和指导决策过程。本文充分利用相关数据，建立优化模型求解决策问题，获得商超收益最大化的补货计划。

2.4 问题四的分析

问题四：要求分析还可以采集哪些数据对本文进行商品的补货、定价决策有所帮助。

为了帮助商超更好地制定蔬菜补货和定价策略，可以收集包括市场竞争情况、顾客消费偏好、供应链信息、节假日因素、促销效果以及商品保存期限等更多相关数据。这些数据可以帮助更准确预测市场需求，如竞争对手价格数据可以考察自身定价策略;购买习惯数据可以判断顾客反应;节假日信息可以评估需求变化;保存期限数据可优化采购和减少损耗。综合利用这些信息，建立更契合实际情况的决策模型，将有助于商超制定出更科学、更合理的蔬菜商品定价和补货计划。

我们绘制如下框架图，便于直观展示文章所建立的模型：

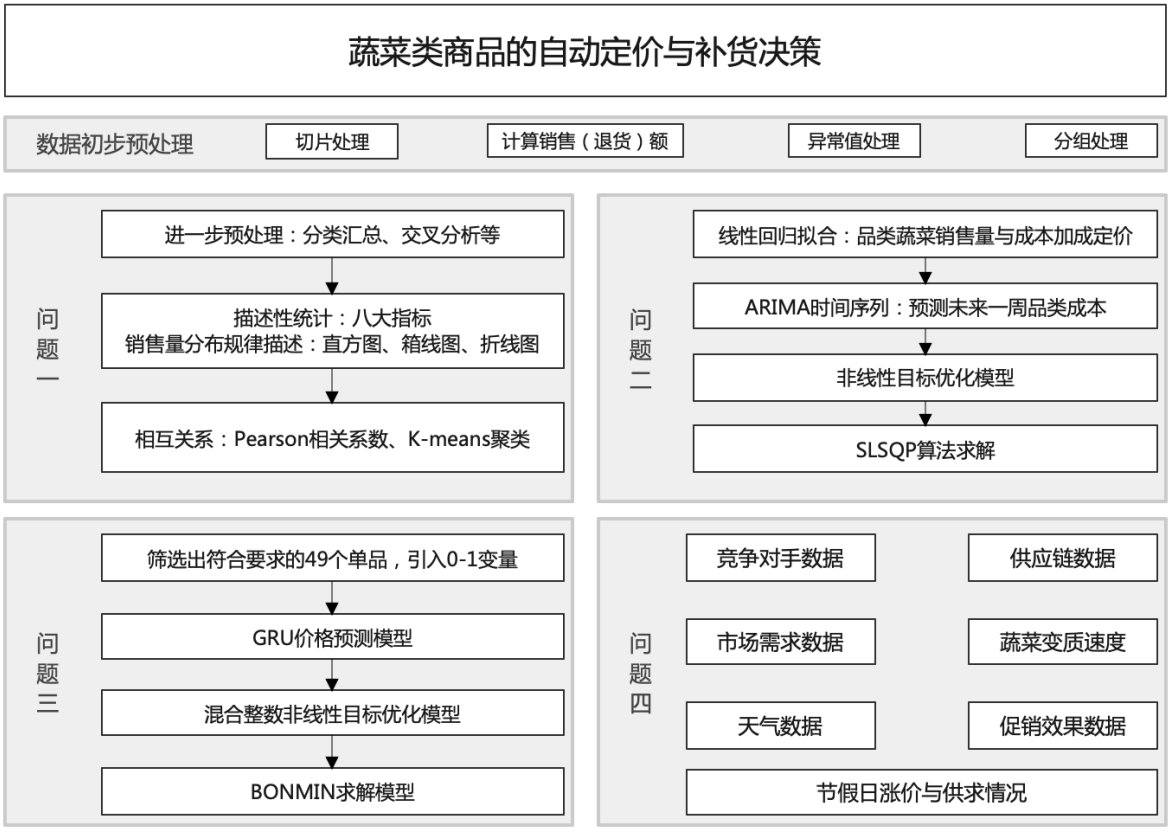

图 1 全文模型框架图

三、 模型假设

1. 假假设在研究期内，除去正常的周期性波动外，市场环境相对稳定，没有发生重大影响供求关系的外部事件。

2. 假设竞争对手在研究期内的定价策略和促销活动保持了相对正常的水平和频率，没有大的波动对我方数据产生影响。

3. 假设所给销售和价格数据集包含了研究期内的全部交易，数据的记录和反映市场情况是准确和可靠的。

4. 假设各种运输、存储、展示等流通环节的损耗率在评估期内维持稳定。

5. 假设在研究周期内，各蔬菜商品不存在隔日继续销售的情况，每日未售出的商品需要及时清理处理。

6. 假设商品采购价格与销售价格呈现相对稳定的关系，不存在价格亏本的情况发生。

四、 符号说明

符号	说明
i	天数
j	对应花菜类、花叶类、辣椒类、茄类、菌类、水生根茎类
num_j	第j类对应的单品数目
h	单品
q_{ij}	第i天第j类的销售量
q_{ih}	第i天h单品的销售量（以下变量同理）
q_h	未来一天h单品的销售量（以下变量同理）
t_{ij}	第i天第j类的退货量
S_{ij}	第i天第j类实际销量
Z_{ij}	第i天第j类的利润
$loss_j$	第j类的平均损耗率
$discount_{ij}$	第i天第j类的平均折扣率
$\cos t_{ij}$	第i天第j类的成本价
p_{ij}	第i天第j类的平均定价
$p_{\min_j}$	过去三年第j类的平均定价最小值

$p_{\max_j}$	过去三年第j类的平均定价最大值
amt_{ij}	第i天第j类的日补货量
w_{ij}	第i天第j类的单日利润率
$need_h$	h单品最小需求量

五、 模型的建立与求解

5.1 数据初步预处理

在解决这一问题之前，我们首先对数据进行了一系列的预处理步骤，以简化数据结构，提取关键信息，并提高算法的运行效率。我们将基于这个预处理过的数据集来建立针对问题一至三的模型。

首先，我们处理了附件 2 中的数据。这个数据集在时间上相对完整，并且没有出现异常离群值，因此我们没有进行删除和清洗操作。为了降低数据量，我们对数据按照月份进行了切片处理（见附录 2）。

接着，我们计算了每笔交易的销售额（退货额）。具体来说，我们将每条交易记录的销量（退货量）与销售单价相乘，得到了销售额（退货额）。

由于附件 2 的数据量极大，我们编写了程序（见附录 1），按照单品编码和销售日期对所有的销售记录进行了分组处理。然后，我们将同一组的销售量和销售额求和，得到了该单品在对应日期的总销售量和总销售额。

由于我们的数据是基于销售记录生成的，因此生成的数据中不存在空值。并且，由于我们在后续的分析中需要研究统计规律，因此在这个阶段，我们并未进行异常值处理。

最后，我们将预处理过的数据和附件 1 进行了右连接，以单品编码为索引，得到了各品类的日总销售量和销售额。通过对数据的观察，我们发现部分单品（例如：赤松茸）在这三年内的销售总量为 0。这意味着这些单品在这三年中没有进行过任何交易，因此对我们的研究没有实际意义。因此，我们剔除了这些数据异常的单品。

5.2 问题一模型的建立与求解

5.2.1 数据预处理

月度，年度处理：首先对各单品与品类三年间的每日总销量进行了月度和年度处理。通过对销售日期字段按月度和年度进行分组，我们得到了单品和品类的月平均和年平均总销量。

合并品类：在处理数据时，本文注意到题目中同名单品括号内的不同数字代表来自不同产地，同时部分同名的单品的括号内还标注有“箱”，“袋”等不同的度量

单位,如果把这些产品按照一个单品进行计算，会导致量纲混乱，并丢失产地与商品的对应关系。因此，我们将名称相同但产地或度量单位不同的单品——例如芜湖青椒（1）与芜湖青椒（2）—视为不同的单品进行处理，而非直接合并。因此，我们后文进行的数据处理，均把名称相同却不同产地，不同量纲的数据视作不同的单品进行处理。

分类汇总：按照附件一中的品类信息对蔬菜数据进行了分类，并创建了一个以品类为键的字典，用于归类每个蔬菜品类对应的商品信息。对于每一个蔬菜品类，我们汇总了相关的数据，如销售量、成本加成定价等，并遍历每个品类下的商品数据，累加销售量并计算平均值等统计指标，从而获得了数据分布和不同类别之间差异的信息。

交叉分析：选择了蔬菜的销售量、成本加成定价等作为交叉分析的变量。根据所选的变量和维度，我们计算了交叉分析的指标，并在数据集中进行了分组、汇总和计算，以得到交叉分析的结果。

5.2.2 研究不同品类销售量分布规律

5.2.2.1 不同品类三年日均销售量

本文首先利用上述预处理后的数据，采用直方图直观的展示了不同品类三年的销售量日均分布图。如图所示，花叶类三年的日均销售量最高，茄类三年的日均销售量最低。

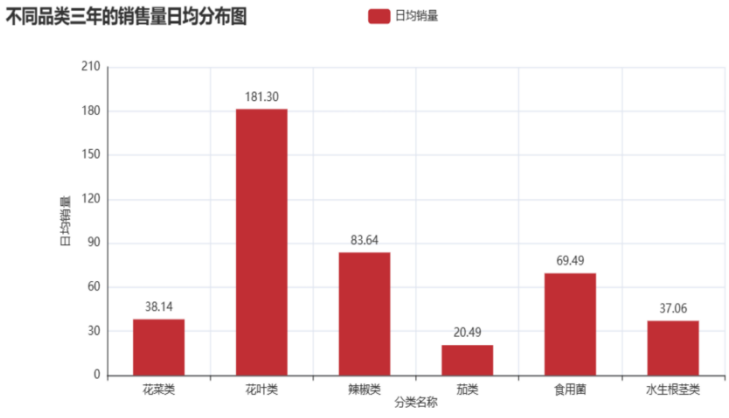

图 2 不同品类三年销售量日均分布

5.2.2.2 不同品类三年月销售量描述性统计

本文采用描述性统计分析对调查总体所有变量的有关数据做统计性描述，主要采用了以下指标进行描述：数据的集中趋势分析（平均值、中位数）、数据离散程度分析（最大值、最小值、标准差、变异系数）、数据的分布（峰度、偏度）、以及一些基本的统计图形。

这 8 个指标的具体解释如下：

1．平均值：指的是算术平均数，也叫均值，是集中趋势的最主要测度值。

2．中位数：将一组数据按由小到大的顺序排列，居于中间位置的变量值。

3．最大值：一组数中的最大值。

4．最小值：一组数中的最小值。

5．标准差：方差的算术平方根，其中方差是各个变量值与其算术平均数的离差平方的算术平均数。

$$\sigma=\sqrt{\frac{\sum\left(x-\bar{x}\right)^2}{n-1}} \tag{1}$$

其中：X 表示数据集，x 表示数据集中的每个数据点，$\bar{x}$ 表示样本的平均值，Σ 表示对所有数据点求和，n 表示样本数据点的个数。

6．变异系数：是测度数据离散程度的相对统计量，主要是用于比较不同样本数据的离散程度。变异系数大，说明数据的离散程度也大；反之，说明数据的离散程度也小。

$$cv=\frac{\sigma}{\bar{x}}\times 100\% \tag{2}$$

其中：σ 表示数据集的标准差。

7．峰度：是分布集中趋势高峰的形状，衡量了数据分布相对于正态分布的尖峰或扁平程度。

$$kurt(X)=\mathbb{E}\left(\frac{X-\mu}{\sigma}\right)^4-3=\frac{1}{n}\cdot\sum\left(\frac{x-\bar{x}}{\sigma}\right)^4-3 \tag{3}$$

其中：正态分布的峰度为 0，当数据分布的峰度大于 0 时，表明数据分布比正态分布更尖峭（尖峰峰度），而当峰度小于 0 时，表明数据分布比正态分布更平坦（平顶峰度）。μ 是 X 的均值

8．偏度：衡量了数据分布的不对称性，即数据分布相对于平均值的左右偏移程度。

$$skew(X)=\mathbb{E}\left(\frac{X-\mu}{\sigma}\right)^3=\frac{1}{n}\cdot\sum\left(\frac{x-\bar{x}}{\sigma}\right)^3 \tag{4}$$

其中：偏度为 0 表示数据分布左右对称，正偏度（$skew(X)>0$）表示数据分

布有右尾较长的倾向，而负偏度（$skew\ (X) < 0$）表示数据分布有左尾较长的倾向。

如图所示，花叶、花菜、水生根茎、茄、辣椒、食用菌这六类蔬菜，变异系数（CV）均大于 0.15，说明当前数据中可能存在异常值。通过对异常的或者表现得较为突出的指标进行分析，本文发现异常数据源于在某几天无人购买该品类或存在有人退货等情况，此均属于市场交易过程的正常现象，因此不对其进行删除清洗。

表 1 不同品类描述性统计分析

变量名	最大值	最小值	平均值	标准差	中位数	方差	峰度	偏度	变异系数（CV）
花叶类	10131.385	1891.076	5514.472	1721.445	5406.397	2963371.461	0.464	0.387	0.312
花菜类	2454.973	436.828	1160.179	445.353	1195.417	198339.456	0.511	0.552	0.384
水生根茎类	2524.01	157.828	1127.26	645.631	1038.302	416839.294	-0.97	0.329	0.573
茄类	1365.551	105.318	623.105	284.235	618.679	80789.332	0.164	0.323	0.456
辣椒类	5182.983	803.319	2544.129	1056.927	2272.862	1117095.036	0.257	0.821	0.415
食用菌	4532.329	762.225	2113.52	969.647	1916.906	940215.401	-0.26	0.682	0.459

5.2.2.3 不同品类三年月销售量

以下是离散趋势分析的结果，采用箱线图形式（见附录 3）展示了花叶类、花菜类、水生根茎类、茄类、辣椒类和食用菌的频数分布情况。在箱线图中，每个箱体代表了对应品类的频数分布。箱体的上边界表示第 75 个百分位数（75%分位数），下边界表示第 25 个百分位数（25%分位数），箱体内的线表示中位数。箱线图的上限和下限（极大值和极小值）表示数据的稳定性范围，而超出上限和下限的点则被视为异常点。

需要注意，这里的极大值和极小值并不是数据的实际最大值和最小值，而是箱线图的内限，用于衡量数据分布的差异和稳定性。如果有数据点超出了极大值或极小值，它们将被视为异常点。在本文中，这些异常点被认为是市场交易过程中的正常现象，可能是由于某品类商品在某一天的销量激增，而采取了打折、批发等促销策略所致。因此，在分析过程中，这些异常点被保留而不被删除或清洗。

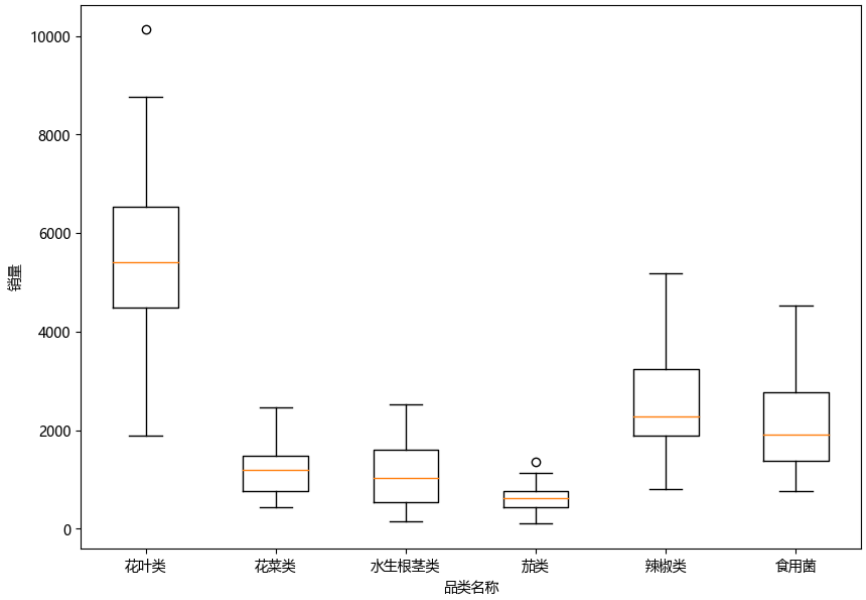

图 3 不同品类三年按月销售量箱线图

5.2.2.4 不同品类三年销量随月份的变化趋势

采用折线图直观描述 6 大品类这三年各月份的销售量趋势变化，如图所示，花叶类蔬菜的销量相比于其他五类遥遥领先，但数据的极差、起伏较大。茄类商品相比于其他五类蔬菜销量最少，同时较平稳。

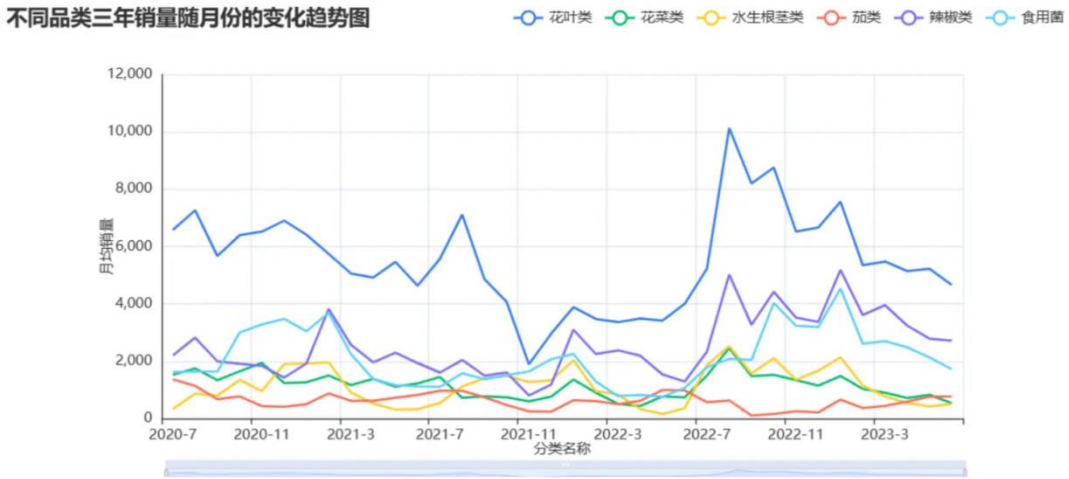

图 4 不同品类三年销量随月份的变化趋势图

5.2.2.5 不同品类年均销售量的季节性规律

由于考虑到蔬菜的季节性生长、季节性需求、价格波动以及节假日和传统习俗等因素的综合影响，猜想蔬菜的销售会存在季节性的规律分布，研究此性质，了解和预测这种季节性分布规律对于蔬菜产业的生产和销售策略制定具有重要意义。因此本文分别采用折线图来动态描述年均销售量的季节性规律分析；绘制堆积柱状图不同品类按年均销售量的每月成分占比.

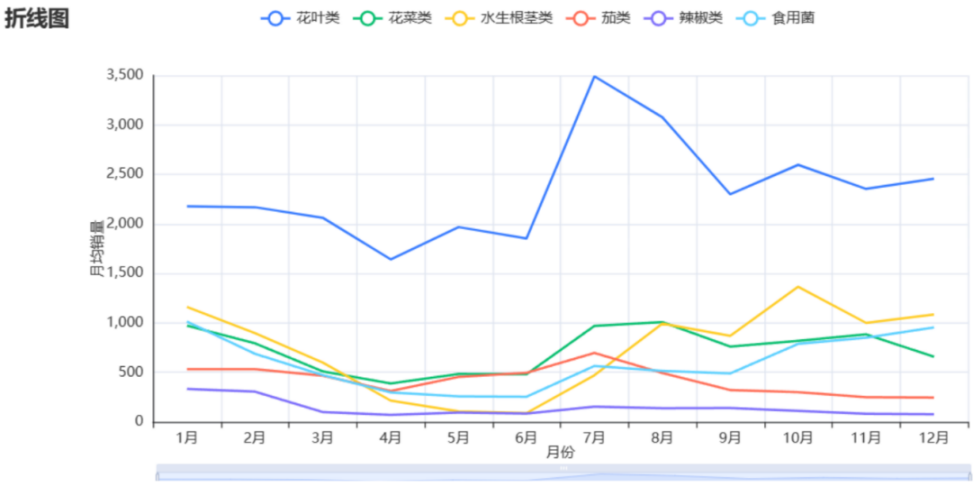

图 5 不同品类按一年均值销售量的季节性规律

从图中可以看出在每个月花叶类的月均销量都是远远高于其他几类的，尤其是经历了 6 月至 7 月的销量激增后，在 7、8 月，花叶类蔬菜销量尤其高；花菜、水生根茎、茄、食用菌这四类蔬菜在后半年的月均销量高于前半年；辣椒类蔬菜月均销量比较平缓，没有出现激增或猛降点。

图 6 不同品类按年均销售量的每月成分占比

从图中可以得到，销量高峰出现在 2 月与 7 月，低谷主要在 11，12 月。除了水生根茎类蔬菜在 10、11、12 月占比明显减少外，其余品类按年均销售量的每月占比变化不大。

5.2.3 研究不同单品销售量分布规律

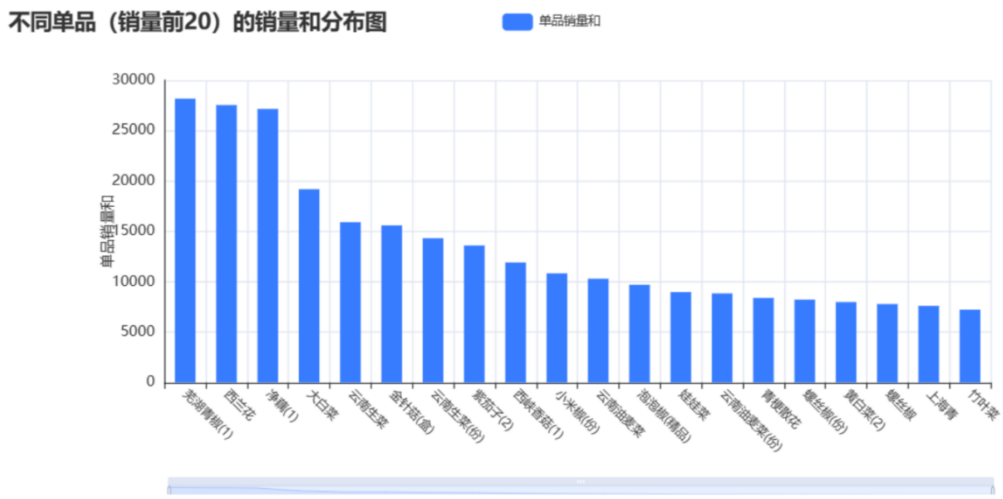

图 7 不同单品（销量前 20）的销量和分布

由于单品种类的数量太大，在此，本文采用直方图仅展示不同单品（三年总销量前 20 个）的销量和，如图所示，单品：芜湖青椒、净藕、西兰花在所研究三年的总销量最高，均超过了 25000 千克。

同时在此本文只展示部分单品（三年总销量前 10）的不同单品箱线图销售量（见附录 4）。

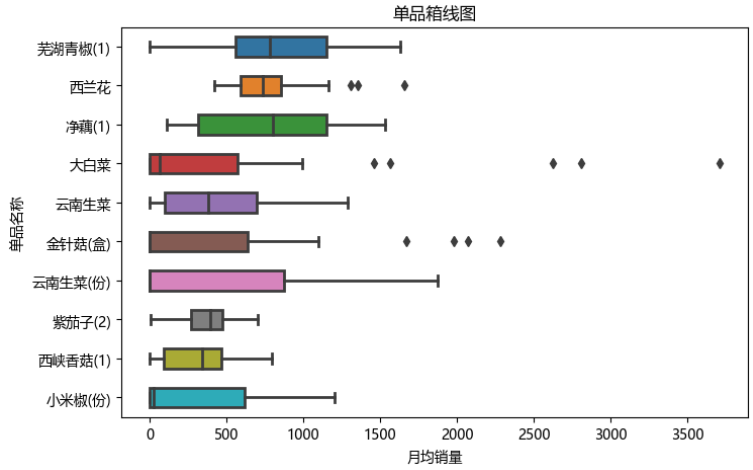

图 8 不同单品（销量前 10）的月销量和箱线图展示

5.2.4 研究分析不同品类销售量相互作用

本文将数据导入 SPSS 进行相关性分析，研究两两品类的相关性。

5.2.4.1 正态分布检验

考虑到有 6 大品类，样本量在 5000 以下，故本文将其视为小数据样本，因此采用 Shapiro-Wilk 检验用于检验数据的正态性。

在 Shapiro-Wilk 检验中，H0（零假设）意为：数据样本服从正态分布；H1（备

择假设）意为：数据样本不服从正态分布。Shapiro-Wilk 检验统计量（W）的计算公式如下：

$$w=\frac{\left(\sum\left(a_i\times y_i\right)\right)^2}{\sum\left(\left(x_i-\bar{x}\right)^2\right)} \tag{5}$$

其中：a_i是 Shapiro-Wilk 检验中的常数，取自已知的方差-协方差矩阵的元素；y_i 是数据样本的排序值（从小到大排序）；x_i是数据样本的原始值。计算得到的 W 值越接近 1，表示数据样本越符合正态分布假设。如果 W 值显著小于 1，根据显著性水平的设定，可以拒绝零假设，即认为数据样本不服从正态分布。

表 2 正态分布检验

变量名	样本量	中位数	平均值	标准差	偏度	峰度	S-W 检验	K-S 检验
花叶类	36	5406.397	5514.472	1721.445	0.387	0.464	0.983(0.854)	0.087(0.929)
花菜类	36	1195.417	1160.179	445.353	0.552	0.511	0.954(0.140)	0.112(0.718)
水生根茎类	36	1038.302	1127.26	645.631	0.329	-0.97	0.948(0.090*)	0.122(0.609)
茄类	36	618.679	623.105	284.235	0.323	0.164	0.979(0.724)	0.078(0.969)
辣椒类	36	2272.862	2544.129	1056.927	0.821	0.257	0.943(0.063*)	0.146(0.392)
食用菌	36	1916.906	2113.52	969.647	0.682	-0.265	0.944(0.068*)	0.131(0.527)

注：***、**、*分别代表 1%、5%、10%的显著性水平

如图所示：采用 S-W 检验，花叶类样本显著性 P 值为 0.854，水平不呈现显著性，不能拒绝原假设，因此数据满足正态分布。花菜类样本显著性 P 值为 0.140，水平不呈现显著性，不能拒绝原假设，因此数据满足正态分布。水生根茎类样本显著性 P 值为 0.090*，水平不呈现显著性，不能拒绝原假设，因此数据满足正态分布。茄类样本显著性 P 值为 0.724，水平不呈现显著性，不能拒绝原假设，因此数据满足正态分布。辣椒类样本显著性 P 值为 0.063*，水平不呈现显著性，不能拒绝原假设，因此数据满足正态分布。食用菌样本显著性 P 值为 0.068*，水平不呈现显著性，不能拒绝原假设，因此数据满足正态分布。

5.2.4.2 Pearson 相关系数

由于数据满足正态分布，因此本文采用 Pearson 相关系数对变量两两之间的相关程度进行分析。

Pearson 相关系数是一种用于衡量两个变量之间线性相关性强弱的统计指标。它

基于变量的原始值进行计算。其理论计算公式如下[1]：

$$\rho = \frac{\sum\left(x_i - \bar{x}\right)\left(y_i - \bar{y}\right)}{\sqrt{\sum\left(x_i - \bar{x}\right)^2 \cdot \sum\left(y_i - \bar{y}\right)^2}} \tag{6}$$

其中：ρ表示 Pearson 相关系数；x_i和y_i分别表示两个变量的第 i 个数据点的原始值；$\bar{x}$ 和$\bar{y}$分别表示两个变量的均值。Pearson 相关系数的取值范围为-1 到 1，其中-1 表示完全负相关，1 表示完全正相关，0 表示无相关性。

表 3 显著性检验表

	花叶类	花菜类	水生根茎类	茄类	辣椒类	食用菌
花叶类	1(0.000***)	0.747(0.000***)	0.484(0.003***)	-0.035(0.842)	0.624(0.000***)	0.546(0.001***)
花菜类	0.747(0.000***)	1(0.000***)	0.472(0.004***)	0.058(0.737)	0.425(0.010***)	0.429(0.009***)
水生根茎类	0.484(0.003***)	0.472(0.004***)	1(0.000***)	-0.466(0.004***)	0.459(0.005***)	0.653(0.000***)
茄类	-0.035(0.842)	0.058(0.737)	-0.466(0.004***)	1(0.000***)	-0.191(0.263)	-0.409(0.013**)
辣椒类	0.624(0.000***)	0.425(0.010***)	0.459(0.005***)	-0.191(0.263)	1(0.000***)	0.576(0.000***)
食用菌	0.546(0.001***)	0.429(0.009***)	0.653(0.000***)	-0.409(0.013**)	0.576(0.000***)	1(0.000***)

注：***、**、*分别代表 1%、5%、10%的显著性水平

对于上表，茄类对花菜类、花叶类和辣椒类的相关性较低，显著性水平大于 0.05，显著性检验不通过，表明不存在显著的相关性。如图所示，本文根据 Pearson 相关系数绘制热力图来直观展现两两变量的相关性大小，可观察得出花叶类和花菜类之间的销量存在较强的相关性，约为 0.75；而茄类与其他品类几乎没有相关性。

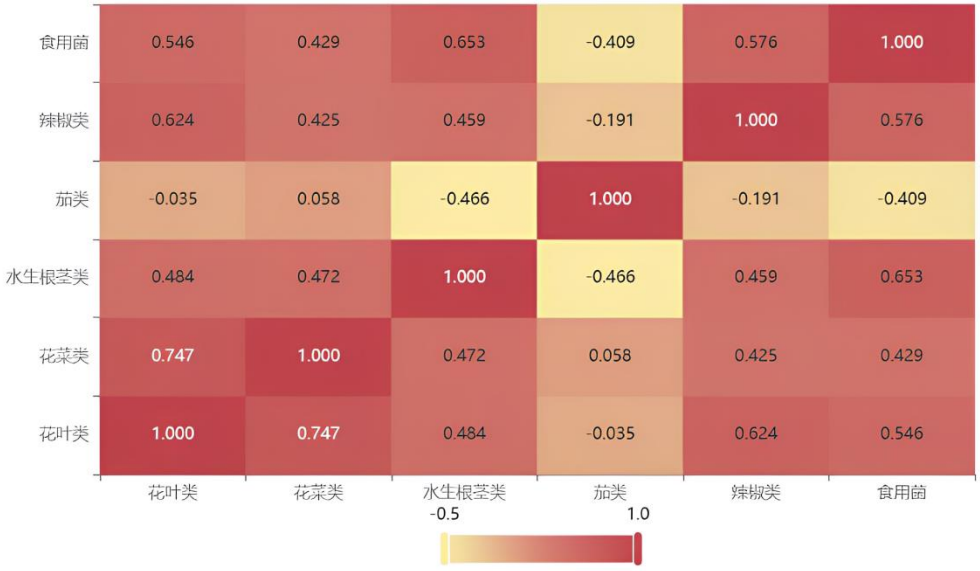

图 9 Pearson 相关系数热力图

5.2.5 研究分析不同单品销售量相互作用

5.2.5.1 K-Means 聚类分析

K-Means 聚类是一种常用的无监督学习算法，用于将数据集中的数据点划分为 K 个不同的簇。其计算步骤如下[2]：

1. 选择 K 个初始聚类中心，可以是随机选择或基于某种启发式方法选择。
2. 初始化每个数据点到最近的聚类中心的簇分配。
3. 重复以下步骤直到收敛：

(1) 更新簇中心：对于每个簇，计算该簇中所有数据点的均值，作为新的聚类中心。

(2) 更新簇分配：对于每个数据点，计算其与所有聚类中心的距离，并将其分配给距离最近的聚类中心所在的簇。根据最终的簇分配和聚类中心，得到聚类结果。

K-Means 聚类的目标是最小化平方误差和（Sum of Squared Errors， SSE），即最小化所有数据点与其所属簇中心之间的距离的总和。计算 SSE 的公式如下：

$$SSE = \sum \sum \| X_i - C_j \|^2 \tag{7}$$

其中：SSE 表示平方误差和；Σ 分别表示对所有数据点和簇中心求和；X_i 表示第 i 个数据点；C_j 表示第 j 个簇中心；$\| \ \|$ 表示欧氏距离。

5.2.5.2 聚类处理与相互作用分析

本文筛选出 2021 年 1 月至 2022 年 12 月的各单品月销量数据。通过研究不同单品在一年中的月销量分布规律，可以对月销量分布规律相似的单品聚类处理，从而

得到在一年中有相同销售模式的单品群体。由于各单品在 1 月至 12 月均有 2021 年和 2022 年的月销量数据，因此本文将两年同一月的数据作均值处理，得到各个单品在一年中的平均每月销量。然后将月份作为指标变量，使用 SPSSPRO 对所有单品进行聚类分析，设置 k=4，将所有单品分成 4 类。聚类类别差异性检验如下表所示。

表 4 字段差异性分析

聚类类别（平均值±标准差）				F	P
类别 1(n=215)	类别 2(n=16)	类别 4(n=12)	类别 3(n=3)		
25.145±76.527	138.12±183.99	289.678±378.54	1021.895±265.464	87.344	0.000***
25.254±66.976	93.456±96.747	296.271±253.004	1019.716±155.549	161.278	0.000***
15.359±36.374	84.986±77.465	281.708±166.575	808.876±357.01	227.828	0.000***
12.439±26.328	69.483±50.292	302.432±121.978	639.39±347.951	283.442	0.000***
11.816±30.524	25.677±50.042	364.962±133.575	673.068±488.609	234.075	0.000***
12.087±32.594	31.695±64.034	371.395±166.941	562.115±373.754	212.352	0.000***
17.456±43.812	103.846±98.81	368.244±220.268	808.702±219.956	218.45	0.000***
27.624±70.424	340.846±213.512	275.028±168.304	1166.203±127.069	218.766	0.000***
18.418±38.628	323.889±182.84	178.838±117.203	787.351±122.592	260.656	0.000***
15.476±35.972	448.55±274.667	140.574±114.405	1079.559±310.727	278.945	0.000***
11.985±32.018	336.215±218.961	64.104±74.244	871.456±181.447	276.035	0.000***
15.033±43.207	377.055±221.736	52.151±56.219	834.578±182.363	248.916	0.000***

注：***、**、*分别代表 1%、5%、10%的显著性水平

聚类类别差异性检验表截取了部分单品的数据。根据该表可知，各变量显著性 P 值为 0.000***，水平上呈现显著性，说明各变量在聚类分析划分的类别之间存在显著性差异，即聚类结果有效。

一共有 246 个单品参与了聚类，在各聚类类别中，聚类 1 的频数最高，占比 87.4%，聚类汇总饼图如下图所示。

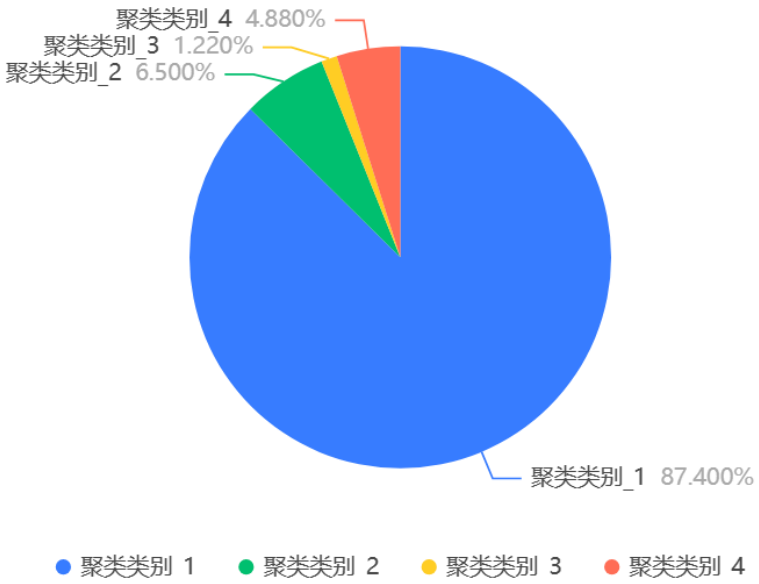

图 10 聚类汇总图

接下来，为了找到具有相同或相似销售模式的单品群体，针对不同的聚类类别进行分析，本文将属于某一聚类类别的各月销量数据均值处理，得到各个聚类群体在不同月份的平均月销量，然后以月份为横轴，月平均销量为纵轴，绘制一年当中每月月均销量的折线图，如下图所示。

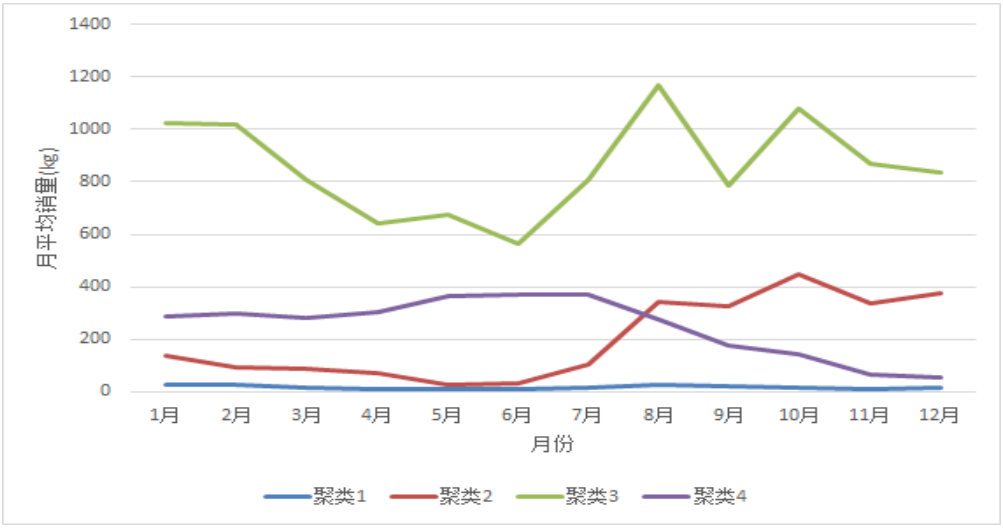

图 11 一年中每月月均销量折线图

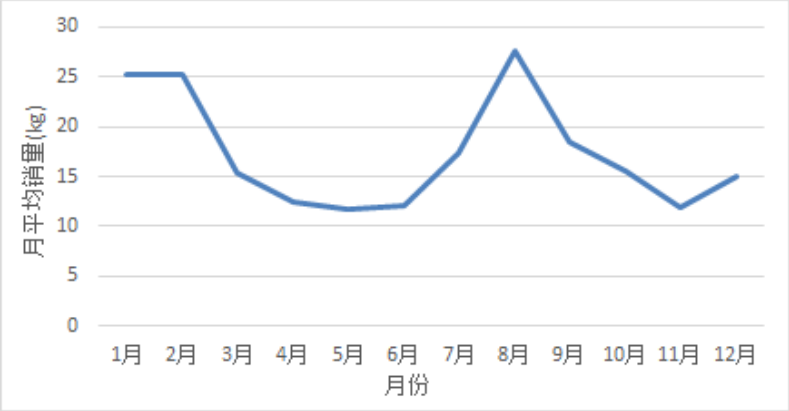

图 12 聚类 1 平均月均销量折线图

根据折线图，可以发现聚类 1 每月月均销量在其他聚类类别中最小，因此单独对聚类 1 绘制折线图。观察折线变化情况，可以方向聚类 1 中的单品在冬季和夏季销量较高，在春季和秋季销量约为冬夏两季的一半。而聚类 2 的单品，在进入夏季后销量不断增加，在秋季达到最大值，进入冬春季后销量逐渐下降。聚类 3 的单品月均销量保持最高，在季节分布上，春季销量最少，其他时间销量较高。对于聚类 4 的单品，其春夏季的销量较高，进入秋季后销量缓慢下降，整体变化趋势较为平缓。

通过对各个聚类群体在不同月份的平均月销量进行分析，我们可以发现各聚类群体在销售模式上的相似性和差异性，进一步探索了单品之间的相互作用。例如，聚类 1 中的单品在冬季和夏季的销量较高，这可能是因为这些单品包含了许多适合在冬季和夏季食用的蔬菜，或者是在这两个季节中有特定的营销策略。

聚类 2 的单品在夏季开始销量增加，秋季达到最大值，然后在冬春季逐渐下降，这意味着这些单品可能受季节影响显著，或者在秋季有特定的促销活动。聚类 3 的单品则在春季销量最少，其他时间销量较高，这可能是因为这些单品在春季供应较少，或者消费者的购买习惯在春季有所改变。

最后，聚类 4 的单品在春夏季的销量较高，秋季后销量缓慢下降，其整体变化趋势较为平缓，这可能是因为这些单品的需求量在全年内较为稳定，或者其销售策略较为稳定。

总的来说，通过聚类分析，对单品进行了分类，并分析了每一类单品在一年中的销售模式，从而揭示出单品之间的相互作用。对于理解商品销售的季节性规律，以及制定针对性的营销策略具有重要的意义。

5.3 问题二模型的建立与求解

5.3.1 研究各品类销售总量与成本加成定价的关系

5.3.1.1 数据预处理

将“各品类按日汇总表”（不同列代表不同的品类，不同行代表各品类每一天的销售量）按照品类的不同拆成六个表，然后根据“交易总表”计算某品类的每日平

均定价，将其作为单独的一列，补充到六个表里，这样就得到了各品类每日销量和每日平均定价的数据，以此来分析每日销量和定价的关系。

5.3.1.2 季节周期性检验

作为数据拟合的一部分，如果数据中存在明显的季节性或周期性模式，且我们未对其进行适当的处理，可能会导致以下问题：拟合模型不准确、误导性结果、错误的决策。因此，在进行数据拟合之前进行季节周期性检验可以帮助我们全面理解数据的特征，并为选择适当的数据拟合模型提供指导，以确保我们能够准确地捕捉到季节性或周期性模式，从而提高预测的准确性和可靠性。

季节分解（Seasonal Decompose）是一种常用的时间序列分析方法，用于将时间序列数据分解为趋势、季节性和残差三个组成部分。它通过将时间序列数据分解为这些组成部分，帮助我们了解数据中的趋势和季节性模式，并检测是否存在周期性变化[3]。通常使用加法模型进行计算，其计算公式如下：

$$\mathrm{Y}(t) = \mathrm{T}(t) + \mathrm{S}(t) + \mathrm{R}(t) \tag{8}$$

其中：$Y(t)$表示在时间点t上的观测值（原始数据）；$T(t)$表示趋势组成部分；$S(t)$表示季节性组成部分；$R(t)$表示残差组成部分。

详细地处理步骤如下：

1．根据时间序列数据，估计趋势组成部分（$T(t)$）。常用的方法包括移动平均、局部回归和指数平滑等。目标是捕捉数据中的长期趋势。

2．从原始数据中减去趋势组成部分，得到去趋势的数据，即：

$$\mathrm{D}(t) = \mathrm{Y}(t) - \mathrm{T}(t) \tag{9}$$

3．对去趋势的数据计算季节性组成部分（$S(t)$）。常用的方法包括移动平均、季节指数和分解方法等。目标是捕捉数据中的季节性模式。

4．计算残差组成部分（$R(t)$）：

$$\mathrm{R}(t) = \mathrm{Y}(t) - \mathrm{T}(t) - \mathrm{S}(t) \tag{10}$$

5．最终的季节分解结果为趋势组成部分（$T(t)$）、季节性组成部分（$S(t)$）和残差组成部分（$R(t)$）。

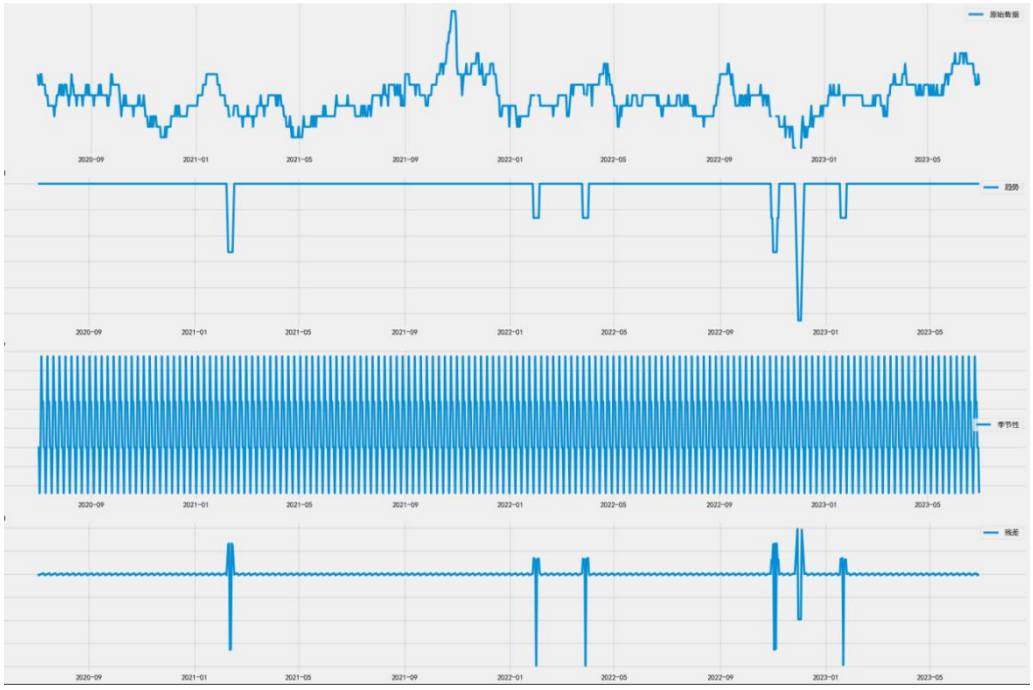

图 13 季节周期性检验图

通过对原始数据进行季节分解检验，发现数据没有较明显的季节性增长趋势，故不需要考虑年度变化对未来的影响。接下来，本文建立回归模型，以品类为单位做补货计划，给出各蔬菜品类未来一周(2023 年 7 月 1 日-7 日)的日补货总量和定价策略，使得商超收益最大。

5.3.1.3 一元线性回归

蔬菜的定价一般采用“成本加成定价”方法：

$$p_{ij} = \cos t_{ij} \cdot (1 + w_{ij}) \tag{11}$$

由于要求分析各蔬菜品类的销售总量与成本加成定价的关系，只有两个变量。于是本文采用一元线性回归。这是一种简单的线性回归模型，用于建立一个自变量（X）和一个因变量（Y）之间的线性关系。它通过拟合一条直线来描述 X 和 Y 之间的关系，它的参数估计是通过最小二乘法进行的。最小二乘法的目标是选择合适的回归系数（β_0 和 β_1），最小化残差平方和。一元线性回归模型的形式可以表示为：

$$Y = \beta_0 + \beta_1 X + \varepsilon \tag{12}$$

其中：Y 表示因变量（要预测的变量）；X 表示自变量（用于预测的变量）；β_0 和 β_1 是回归系数，分别表示截距和斜率，用来确定直线的位置和斜率；ε 是误差项，表示模型无法解释的随机误差。

通过拟合实际销量和定价的关系，本文得到六个品类销量与定价的回归方程，如下（代码见附录 5）：

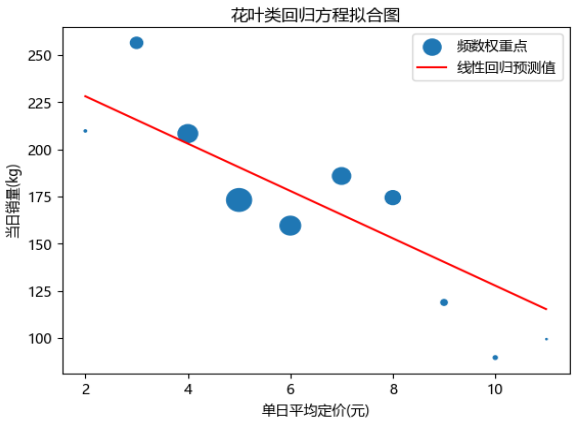

图 14 花叶类回归方程拟合

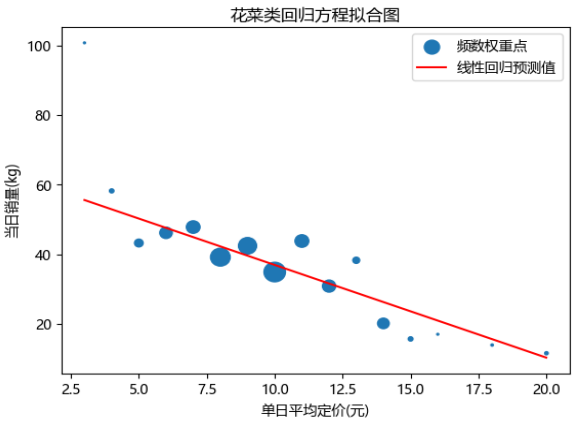

图 15 花菜类回归方程拟合

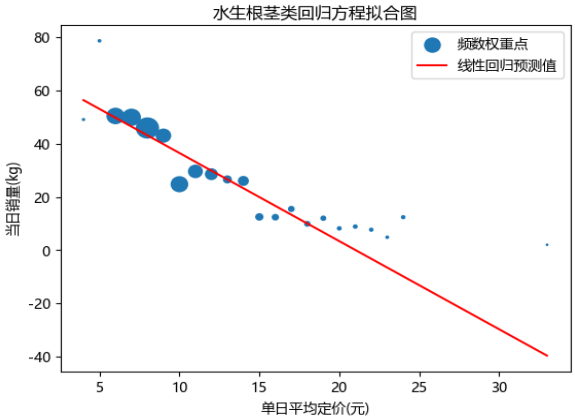

图 16 水生根茎类回归方程拟合

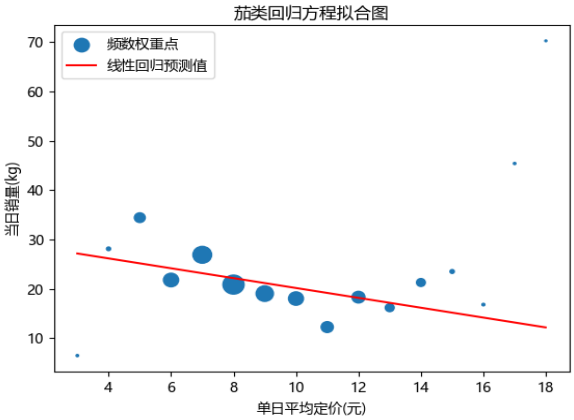

图 17 茄类回归方程拟合

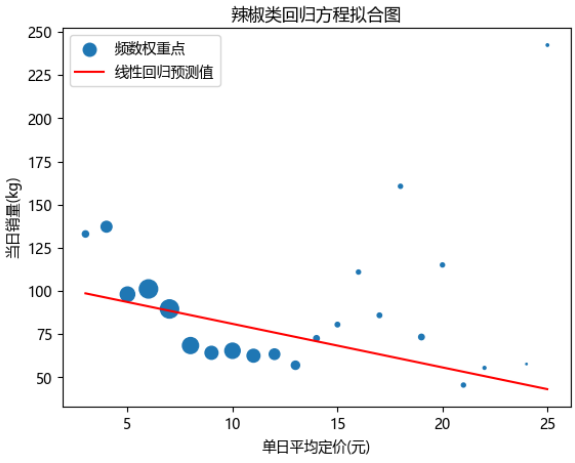

图 18 辣椒类回归方程拟合

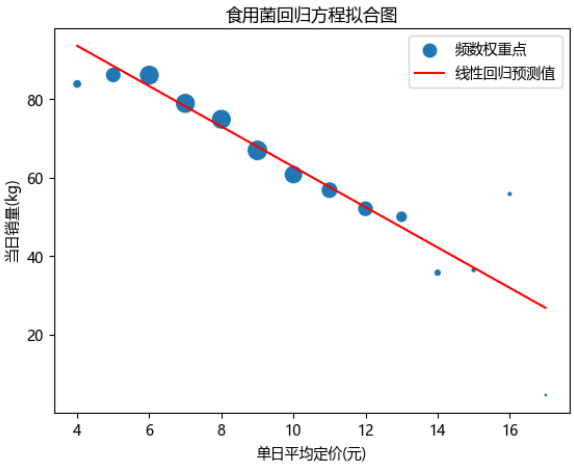

图 19 食用菌回归方程拟合

可以直观的看到，除了辣椒类回归方程有相对大的偏差外，大部分回归方程的拟合效果极佳，十分契合数据。

5.3.1.4 模型检验

在一元线性回归中，本文通过确定系数（R^2）和 F 显著性检验用于评估拟合的直线对数据的拟合程度和模型的可靠性。

R^2 衡量了回归模型对因变量变异性的解释程度，取值范围在 0 到 1 之间，越接近 1 表示模型拟合得越好，公式如下：

$$\mathbf{R}^2 = 1 - (\mathrm{SSR}/\mathrm{SST}) \tag{13}$$

其中：SSR 是残差平方和，表示模型预测值与实际值之间的差异的平方和。SST 是总平方和，它衡量了因变量与其均值之间的差异的平方和。

$$\mathrm{SST} = \Sigma\left(Y - \overline{Y}\right)^2 \tag{14}$$

Y 表示每个观测值的因变量值；$\overline{Y}$ 表示因变量的均值。

F 显著性检验是用于评估线性回归模型的整体拟合程度的统计检验方法。它通过比较模型的回归平方和与残差平方和之间的比值，判断模型是否具有统计显著性，其公式如下：

$$\mathrm{F} = \frac{(SST/k)}{(SSR/(n-k-1))} \tag{15}$$

其中：n 表示样本数量，$k+1$ 表示总体回归系数的个数（包括截距项）

在线性回归中，F 显著性检验通常用于评估回归模型中自变量的整体显著性，即判断自变量是否对因变量的解释具有统计意义。F 检验的原假设（H_0）是回归模型中所有自变量的系数均为零，即自变量对因变量没有显著影响。备择假设（H_1）是至少有一个自变量的系数不为零，即自变量对因变量具有显著影响。

我们设定一个显著性水平（为 0.05），将计算得到的 p 值与该显著性水平进行比较。如果 p 值小于显著性水平，我们拒绝原假设，认为模型具有统计显著性，即自变量的线性组合对因变量的解释有统计意义。相反，如果 p 值大于显著性水平，我们接受原假设，认为模型在统计上不具有显著性，即自变量的线性组合对因变量的解释不具有统计意义。

5.3.1.5 结果展示

表 5 不同品类拟合方程及效果展示

品类名称	回归方程	R 方拟合度	F 显著性检验
花菜类	$y=63.621-2.664x$	0.66	P=0.000***
花叶类	$y=253.26-12.542x$	0.481	P=0.000***
辣椒类	$y=106.217-2.523x$	0.154	P=0.000***
茄类	$y=30.149-0.999x$	0.203	P=0.000***
食用菌	$y=114.208-5.14x$	0.953	P=0.000***
水生根茎类	$y=69.628-3.311x$	0.813	P=0.000***

*上表中的 y 为该品类的日平均定价，x 为该品类的日平均销量

如图所示，说明模型具有统计显著性，食用菌和水生根茎类的拟合效果良好，花叶类和花菜类的拟合效果较好。由于茄类和辣椒类存在一部分异常值，拟合效果较差，但模型仍具有显著的线性相关性。

5.3.2 最优日补货总量和定价策略

考虑以品类为单位做补货计划，给出各蔬菜品类未来一周(2023 年 7 月 1-7 日)的日补货总量和定价策略，使得商超收益最大

5.3.2.1 模型准备

计算各品类的平均损耗率，损耗产品的平均折扣率，作为下面目标优化可直接使用的变量。

$$loss_j=\sum_{h=1}^{nvm_j}\frac{q_{ih}\cdot loss_n}{q_{ih}} \tag{16}$$

表 6 各品类平均损耗率

分类名称	花叶类	花菜类	水生根茎类	茄类	辣椒类	食用菌
平均损耗率	9.065708	11.5532	16.10415	3.178154	6.994084	9.627786

$$discount_{ij}=P_{in_{折}}/P_{in_{不折}} \tag{17}$$

过去平六月平均折扣率 = 23 年 6 月各品类中打折单品的平均价格除以不打折单品的平均价格（公式 15）。

表 7 各品类平均折扣率

	花菜类	花叶类	辣椒类	茄类	食用菌	水生根茎类

打折前	13.98844	5.636594	7.11591	8.272471	11.53505	16.09311
打折后	9.164706	3.640902	6.241818	Null	2.584737	11.65957
比值	0.655163	0.64594	0.877164	Null	0.224077	0.724507
折扣率	70%	65%	90%	100%	25%	75%

5.3.2.2 ARIMA 时间序列预测模型

本文在前文已做了季节周期性检验，验证了此时间序列数据不具有季节性。所以我们选择 ARIMA（Autoregressive Integrated Moving Average）模型，这是一种常用的时间序列预测模型，它结合了自回归（AR）和移动平均（MA）的概念，以及差分（I）的操作，基于时间序列数据的特征和模式，用于预测未来预测未来一周(2023 年 7 月 1-7 日)的各品类的每日成本（核心代码见附录 6）。

ARIMA 模型的理论原理如下：

1．自回归（AR）部分：

AR 表示自回归，它基于过去观测值的线性组合来预测当前观测值。AR 模型假设当前观测值与过去若干个观测值之间存在关联关系，即当前观测值可以由过去的观测值加权组合得出。AR(p)模型的数学公式为：

$$Y(t) = c + \phi_1 \cdot Y(t-1) + \phi_2 \cdot Y(t-2) + ... + \phi_p \cdot Y(t-p) + \epsilon(t) \quad (18)$$

其中：$Y(t)$表示当前观测值；$\phi_1, \phi_2, ..., \phi_p$表示自回归系数，表示过去观测值的权重；$c$表示常数项；$\epsilon(t)$表示误差项，表示模型无法解释的随机部分。

2．移动平均（MA）部分：

MA 表示移动平均，它基于过去观测值的误差项的线性组合来预测当前观测值的误差项。MA 模型假设当前观测值的误差项与过去若干个观测值的误差项之间存在关联关系。$MA(q)$模型的数学公式为：

$$Y(t) = c + \epsilon(t) + \theta_1 \cdot \epsilon(t-1) + \theta_2 \cdot \epsilon(t-2) + ... + \theta_q \cdot \epsilon(t-q) \quad (19)$$

其中：$\epsilon(t)$表示当前观测值的误差项；$\theta_1, \theta_2, ..., \theta_q$表示移动平均系数，表示过去误差项的权重；$c$表示常数项。

3．差分（I）部分：

差分操作用于处理时间序列数据的非平稳性。如果时间序列数据不平稳（即存在趋势或季节性），可以对数据进行差分操作，将其转化为平稳的时间序列。差分操作是通过计算观测值与之前观测值之间的差异来实现的。I(d)表示差分操作，其中 d 表示差分的次数。通过对时间序列数据进行 d 次差分操作，可以将非平稳时间序列转

化为平稳时间序列。

综合自回归、移动平均和差分的概念，ARIMA 模型可以表示为 ARIMA(p， d，q)，其中 p、d 和 q 分别表示自回归阶数、差分阶数和移动平均阶数。ARIMA 模型根据历史观测值和模型参数，通过迭代算法进行参数估计和模型拟合，然后可以用于未来观测值的预测。ARIMA 模型是通过寻找历史数据之间的自相关性，来预测未来（假设未来将重复历史的走势），要求序列必须是平稳的。

5.3.2.3 数据检测与参数选取

由于数据较多，本文用花菜类为例，按照如下步骤进行处理：

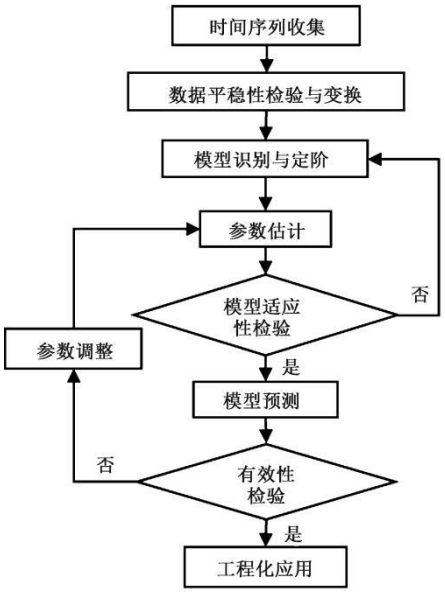

图 20 ARIMA 数据处理步骤[4]

1．平稳性检测（adfuller 单位根检验）

ADF 检验是一种常用的单位根检验方法，用于检验时间序列数据是否具有单位根（即非平稳性）。ADF 检验的原假设（H0）是数据具有单位根，即非平稳性。备择假设（H1）是数据不具有单位根，即平稳性。对于一个一阶差分的自回归模型，可以表示为：

$$\Delta Y(t) = \alpha + \beta \cdot t + \gamma \cdot Y(t-1) + \phi_1 \cdot \Delta Y(t-1) + \epsilon(t) \tag{20}$$

其中：$\Delta Y(t)$ 表示观测值的一阶差分；α 和 β 是常数项和时间趋势项；γ 是 $Y(t-1)$ 的系数，用于检验是否存在单位根；ϕ_1 是差分项的系数；$\epsilon(t)$ 是误差项。

ADF 检验统计量的临界值取决于所选的显著性水平（本文为 0.05 ）和样本量。

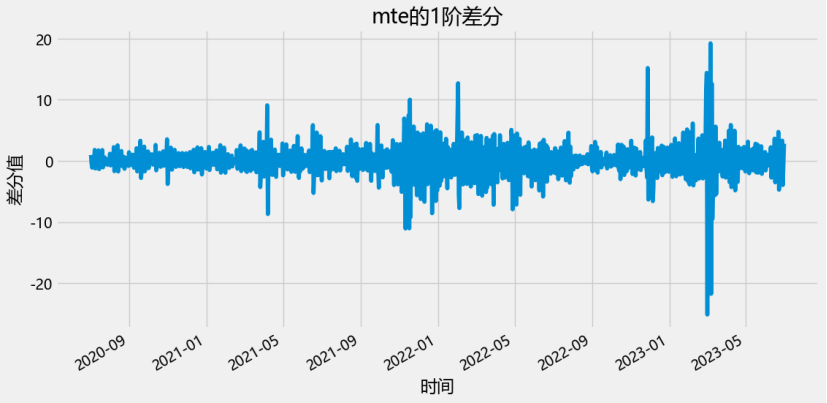

图 21 一阶差分示意图

原始数据的 P 值>0.05 所以不满足平稳性要求。一阶差分的 P 值<0.05，且 T 值小于 1%，5%，10%下的统计值，因此可以极显著的拒绝原假设，说明数据是平稳的。如图所示一阶差分数据已经平稳就无需继续做二阶差分了。

2．白噪声检测（基于自相关函数（ACF）和偏自相关函数（PACF））

白噪声检验是用于检验时间序列数据是否符合白噪声假设的一种统计检验方法。白噪声是一种具有无序随机性、均值为 0、方差为常数、无自相关性的时间序列。白噪声检验的原假设（H0）是数据是白噪声，即具有无序随机性、均值为 0、方差为常数、无自相关性。备择假设（H1）是数据不是白噪声。

计算公式如下：

• 自相关函数（ACF）：

$$ACF(k) = \text{Corr}(Y(t), Y(t-k)) \tag{21}$$

其中，$Y(t)$是时间序列数据中的观测值，$Y(t-k)$是与之前 k 个时间步长相关的观测值；Corr 表示相关系数。

• 偏自相关函数（PACF）：

$$PACF(k) = \text{Corr}(Y(t), Y(t-k) | Y(t-1), Y(t-2), ..., Y(t-k+1)) \tag{22}$$

其中，$Y(t)$是时间序列数据中的观测值，$Y(t-k)$是与之前k个时间步长相关的观测值；$Y(t-1), Y(t-2), ..., Y(t-k+1)$是中间的观测值，通过条件化来消除了这些观测值对相关性的影响。

对于 ACF 和 PACF 的值，可以根据所选的显著性水平（本文为 0.05 ）绘制置信区间。观察 ACF 和 PACF 是否在置信区间范围内，如果没有显著的自相关性，可以接受原假设，认为数据是白噪声。

3．以 BIC 最小为标准确定 p， q 值

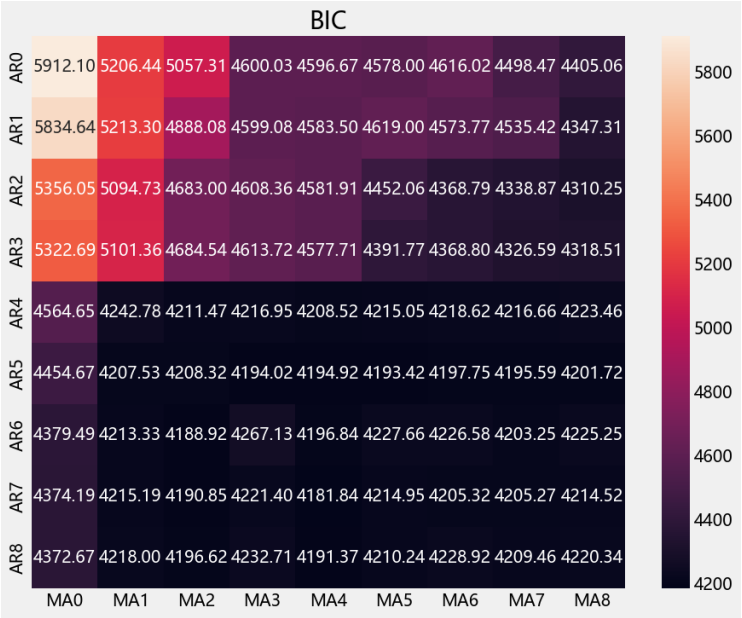

图 22 BIC 数值示意图

这里给出的 AIC 标准下和 BIC 标准下的 p， q 选择一致，所以我们可以直接确定 p=6， q=2 为模型最佳参数。

5.3.2.4 结果展示

表 8 未来一周各品类成本预测结果表

预测日期	花菜类	花叶类	辣椒类	茄类	食用菌	水生根茎类
2023-07-01	9.751429	6.464566	3.630144	18.7988	5.001102	7.155034
2023-07-02	17.51806	6.115882	7.079846	17.69845	21.84939	14.03318
2023-07-03	12.28033	5.429157	3.479066	11.33799	23.34437	10.88586
2023-07-04	10.40713	6.616787	4.677194	18.22341	10.45993	12.38348
2023-07-05	11.10159	6.533293	6.892696	16.19046	6.8643	12.64233
2023-07-06	10.23595	5.744488	3.023501	10.90409	5.484994	7.687106
2023-07-07	12.39064	6.5933	3.751612	16.81247	6.502647	8.894771

5.3.2.5 模型检验

如图所示，是对原数据集拟合的效果展示（此部分代码见附录 7）（蓝线为原始数据，红线为预测数据），且展示了残差分析图，不难看出，该模型对数据集拟合的效果相当好，能提取出原数据集中的绝大部分信息，十分契合本题数据。

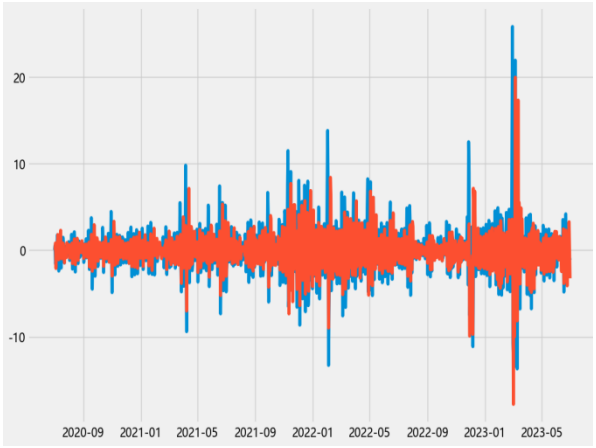

图 23 拟合效果比对图

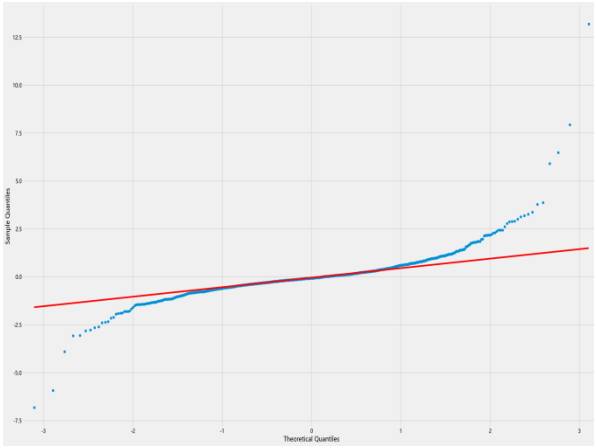

图 24 残差分析图

5.3.3 非线性目标规划

5.3.3.1 模型建立

对于各蔬菜品类未来一周的日补货总量和定价策略建立模型，本文认为最大化每天商超的盈利即可最大化未来一周商超的总盈利。因此以求取未来第一天商超盈利最大化为例展示模型。

决策变量：p_{ij}，分别表示第 i 天第 j 类的定价；amt_{ij}，分别表示第 i 天第 j 类的日补货量。于是在第 1 天，花菜类的盈利可表示为

$$\begin{aligned} Z_{11} &= \{Z_{11}\}_{\text{不折}} + \{Z_{11}\}_{\text{折}} \\ &= S_{11} \cdot (1 - loss_{11}) \cdot p_{11} + S_{11} \cdot loss_{11} \cdot discount_{11} \cdot p_{11} - \cos t_{11} \cdot amt_{11} \end{aligned} \tag{23}$$

其中：

$$S_{ij} = q_{ij} - t_{ij} \tag{24}$$

商超在第一天的总盈利为

$$Z_{\text{总}} = \sum_{j=1}^{6} Z_j \tag{25}$$

对于等式约束，为前文拟合出的各品类销售量与定价的函数关系。

对于不等式约束，本文考虑了其定价需满足合理范围；销售量S_{ij}不能超过日补货量amt_{ij}，退货量不能超过销售量；以及相关变量的非负约束。

综上所述，问题二的非线性优化模型为：

$$\max Z_{总i}=\sum_{j=i}^{6}Z_{ij}$$

$$st.\begin{cases}S_{ij}=m_j+k_j\cdot p_{ij}\\ p_{\min_j}\le p_{ij}\le p_{\max_j}\\ S_{ij}\le \mathrm{ant_{ij}}, t_{ij}\le S_{ij}\\ p_{ij},S_{ij}\ge 0\end{cases} \tag{26}$$

其中变量说明如下：

k_j指第 j 类蔬菜销量与定价拟合函数的斜率，m_j指第 j 类蔬菜销量与定价拟合函数的截距，$Z_{总i}$指第 i 天的利润。其余变量见符号说明。

5.3.3.2 SLSQP 模型求解

SLSQP（Sequential Least Squares Programming） 是一种求解优化问题的算法，特别适合处理带有约束的问题。这是一种基于梯度的优化算法，也是一种迭代算法，在每一步都会解一个二次规划（Quadratic Programming，QP）子问题[5]。这种算法的主要思想是，通过在每一步求解一个二次规划问题，来逼近原始的优化问题。这个二次规划问题是在当前点的邻域内对原始问题的线性近似，因此可以通过解这个二次规划问题来得到下一个迭代点（代码见附录 8）。

SLSQP 的主要步骤如下：

1．选定一个初始点，计算目标函数和约束条件在该点的值和梯度。

2．在当前点，构建一个二次规划问题，这个问题是对原始问题在当前点的线性近似。

3．解这个二次规划问题，得到一个新的点。

4．检查是否满足停止条件（例如函数值变化小于某个阈值，或者迭代次数达到上限）。如果满足则停止，否则回到步骤 2。

5.3.3.3 结果展示

表 9 最优日补货总量和定价策略

时间	花菜类		花叶类		辣椒类		茄类		食用菌		水生根茎类	
	日补货量	利润率	日补货量	利润率	日补货量	利润率	日补货量	利润率	日补货量	利润率	日补货量	利润率
2023/7/1	18.35524	0.742478	84.76223	1.078203	48.49682	5.302119	5.684499	0.302689	50.83716	4.449545	22.47195	0.99052
2023/7/2	10.341	0.141679	87.02049	1.167247	44.11436	2.476714	6.234124	0.352594	28.67036	0.621229	10.60751	0.270245
2023/7/3	14.86581	0.49031	91.4681	1.376061	48.68874	5.553923	9.411174	0.830888	28.4245	0.611951	16.03647	0.486874
2023/7/4	17.45049	0.665327	83.77641	1.042273	47.16667	4.004028	5.971906	0.328033	42.16136	1.598402	13.45316	0.370061
2023/7/5	16.49226	0.593553	84.31711	1.061774	44.35212	2.557442	6.987365	0.432005	46.06991	2.32792	13.00665	0.352676
2023/7/6	17.68669	0.684515	89.4258	1.273979	49.26748	6.465565	9.627908	0.883847	47.61148	2.78085	21.55415	0.888804
2023/7/7	14.7136	0.481653	83.92851	1.047708	48.3425	5.114375	6.676671	0.397524	43.53298	1.806551	19.471	0.70309

表 10 每日收益最大值表

时间	花菜类	花叶类	辣椒类	茄类	食用菌	水生根茎类
2023/7/1	122.0862	554.6699	925.6804	32.34588	466.498	146.378
2023/7/2	18.49744	584.6189	765.9399	38.90321	148.3726	32.61529
2023/7/3	80.07982	645.9051	933.0217	88.65885	145.8387	74.54386
2023/7/4	110.3473	541.8423	875.5982	35.69937	320.8608	52.46174
2023/7/5	98.56132	548.8597	774.2182	48.87215	383.1086	49.03717
2023/7/6	113.3546	617.3844	955.3344	92.78938	409.1764	134.6654
2023/7/7	78.44836	543.8118	919.799	44.62256	342.0771	109.8931

5.4 问题三模型的建立与求解

5.4.1 筛选可售单品并分析各品类蔬菜商品最小需求量

从上表可知，花叶类商品的最小需求量最大，其次是辣椒类。花叶类与辣椒类的总和在六大类别的需求量中占比超过 70%。需求量相对较小是花菜类和茄类，占比小于 10%。

为了制定 7 月 1 日的补货策略，本文选择了 2023 年 6 月 24 日至 6 月 30 日一周内有销售记录的 49 个可售蔬菜品种进行分析。在寻找最优补货方案时，需考虑不同蔬菜品类的市场需求量。为合理评估各品类的最小需求，本文采用最近一周该品类日销量的最小值作为补货下限。

具体来看，先统计 6 月 24 日至 6 月 30 日每日中各品类商品的销量，从中选择销量最低的一天，作为该品类在 7 月 1 日的最小补货量。经统计，不同蔬菜品类的最小需求量情况如下表所示。

表 11 不同蔬菜品类的最小需求量表

类别	花叶类	花菜类	水生根茎类	茄类	辣椒类	食用菌
数量	80.524	8.083	10.384	8.415	67.12	35.271

从结果可以看出，花叶类商品需求量最大，达到了 80.52kg。辣椒类次之，为 67.12kg。这两个品类合计占所有品类总需求量的 70%以上。相对而言，花菜类和茄类商品需求规模较小，均低于 10kg，合计占比不足 10%。

上述分析基于近一周销量数据确定了不同品类的市场容量下限。这为后续制定平衡各品类需求的补货策略奠定了基础。本文将在满足最小需求量的前提下，优化库存成本和销售收入，寻找最佳的补货方案。

5.4.2 利用 GRU 预测可售单品在 7 月 1 日的批发价格（成本）

为了预测 7 月 1 日各可售蔬菜品种的批发价格，本文构建了基于门控循环单元(GRU)的循环神经网络(RNN)模型（见附件 9）。GRU 是当前深度学习领域常用的 RNN 结构之一。

相较于简单的 RNN，GRU 设计了更新门和重置门机制，可以更有效地捕捉时间步间的长程依赖关系，克服了 RNN 中的梯度消失和爆炸问题，因而更适合处理较长的时间序列预测任务[6]。

具体而言，本文的 GRU 价格预测模型包含输入层、多个隐藏层和输出层。隐藏层由若干 GRU 单元组成，每个 GRU 单元内部包含更新门、重置门和记忆单元。这种结构可以学习时间序列中价格变化的时序规律。相比线性回归等传统方法，GRU 模型可以更好地提取时间特征，对价格变动模式进行建模。如图所示，为基于 GRU 的 RNN 模型示意：

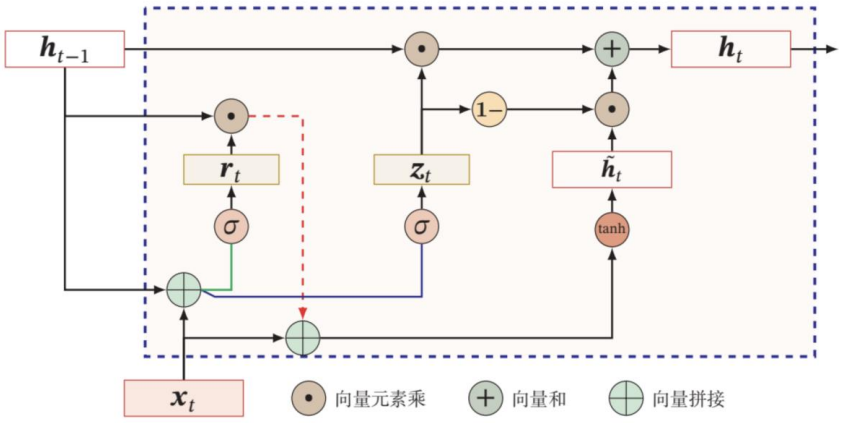

图 25 GRU 模型展示图

本文基于以下考虑选择 GRU 模型:

1) 批发价格具有时间相关性和顺序模式;

2) GRU 对长序列数据的记忆能力强，可以利用近 3 个月的历史数据进行模型训练;

3) 与 LSTM 相比，GRU 训练效率更高，模型结构更简单;

4) GRU 支持变长序列输入，适合处理不同单品时间范围不完全一致的情况。

研究采用近 3 个月共 90 天的历史批发价格数据作为模型输入，通过 GRU 网络进行时序特征学习，最终输出 7 月 1 日价格的预测。相较传统统计方法，本文所构建的 GRU 递归神经网络模型可以更好地挖掘时间序列的内在规律，实现对批发价格的准确预测。

GRU 的基本公式如下：

1) 更新门z_t：

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t] + b_z) \tag{27}$$

2) 重置门r_t：

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t] + b_r) \tag{28}$$

3) 候选隐藏状态$\tilde{h}_t$：

$$\tilde{h}_t = \tanh(W \cdot [r_t \odot h_{t-1}, x_t] + b) \tag{29}$$

4) 最终的隐藏状态h_t：

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \tag{30}$$

其中，σ是 sigmoid 函数，$\odot$表示逐元素乘法，W_z、W_r、W以及b_z、b_r、b是需要学习的参数，h_{t-1}是前一时间步的隐藏状态，x_t是当前时间步的输入。

本文使用 Python 代码实现 GRU 预测未来一日批发价格的任务，部分预测结果如下表：

表 12 未来一日批发价格（部分结果展示）表

单品名称	云南生菜(份)	小米椒(份)	云南油麦菜(份)	金针菇(盒)	芜湖青椒(1)	竹叶菜	西兰花
Predictions	3.38	2.01	2.81	1.53	3.59	2.21	7.58
单品名称	小皱皮(份)	螺丝椒(份)	紫茄子(2)	娃娃菜	双孢菇(盒)	苋菜	海鲜菇(包)
Predictions	2.08	4.09	3.58	4.34	3.45	2.30	2.17
单品名称	菠菜(份)	姜蒜小米椒组合装(小份)	螺丝椒	奶白菜	净藕(1)	木耳菜	小青菜(1)
Predictions	4.03	2.30	8.71	2.56	10.34	3.15	2.70
单品名称	西峡花菇(1)	红薯尖	长线茄	洪湖藕带	上海青	枝江青梗散花	高瓜(1)
Predictions	14.62	2.65	6.97	18.25	4.16	8.36	11.56
单品名称	青茄子(1)	虫草花(份)	青红杭椒组合装(份)	红椒(2)	菱角	云南生菜	菜心
Predictions	4.00	2.77	3.28	12.63	9.25	5.50	4.57
单品名称	蟹味菇与白玉菇双拼(盒)	圆茄子(2)	七彩椒(2)	菠菜	外地茼蒿	高瓜(2)	云南油麦菜
Predictions	3.35	3.35	12.02	9.51	9.25	13.23	2.89
单品名称	红莲藕带	白玉菇(袋)	鲜木耳(份)	木耳菜	野生粉藕	青线椒	紫茄子

				(份)		(份)	(1)
Predictions	5.65	3.31	1.42	1.86	12.61	2.52	4.63

使用混合整数非线性规划（MINLP）模型(代码见附录 10)

5.4.2.1 建立可售单品定价与销量的回归方程

对于各可售单品的定价与销量之间的关系，本文仍采用线性回归模型进行拟合。在模型构建前，我们首先对原始数据进行了异常值检测与处理。具体来说，使用 3σ 原则识别离群值，即超出样本均值正负 3 倍标准差之外的异常数据点。这部分异常数据点在后续分析中被移除，不参与模型参数估计。

线性回归可以较为直观地反映出定价与销量之间的负相关特点，从而简化问题的建模过程。然而，在具体的数据处理中，部分单品的历史交易样本较少，直接建立线性回归模型容易受剩余数据点的影响，无法很好地反映该单品的整体统计规律。为了提高模型的鲁棒性，本文对各单品的线性回归方程进行了修正。

具体的修正方法是:首先，在三年的时间跨度内，分别对 49 个可售单品的定价和销量进行线性回归，得到待修正的原始回归方程。其次，依据问题二中得到的 6 个蔬菜品类级别的回归方程，考量各单品与其所属品类整体回归趋势的关系，综合两个层次的信息，构建单品回归方程的修正格式:

$$\hat{y}_i = w\hat{y}_i^{o} + (1-w)\hat{y}_c \tag{31}$$

其中，权重超参数 w 取值为该单品三年有售天数占总天数的比例，再乘以缩放系数 0.4，即:$w = 0.4 * \frac{有售天数}{总天数}$。缩放系数 0.4 用于控制单品个体信息在修正过程中的作用力度。

上述修正格式充分利用了品类层次的统计规律来约束单品模型，使其在样本较少时也能反映较好的趋势关系。相比直接建模，这种方法可以获得更稳健、更准确的单品定价-销量关系方程，为后续定价决策提供良好支持。

5.4.2.2 建立模型目标函数与约束条件

为了制定 7 月 1 日各单品的具体补货与定价计划，本文将问题转换为一个整数规划模型进行建模和求解。根据题设，需使可售单品总数控制在 27-33 个之间，每个单品的订购量满足最小陈列量 2.5 千克的要求。同时，补货计划应在满足市场对各品类商品需求的前提下，最大化商超的总收益。

由于可售单品总数需要控制在 27-33 个之间，因此对于 49 个可售单品引入 0-1 整数决策变量b_h表示是否选择该单品补货，$b_h = 1$表示该单品被选择，$b_h = 0$表示不被选择。

建立目标函数：

本文考虑了各单品在 7 月 1 日的损耗和折扣情况：

1) 损耗情况根据附件 4 的数据，其反映了近期商品的损耗率，将其作为 7 月 1 日的损耗情况。

2) 折扣情况采用问题二的分析结果，使用各单品所属品类的平均折扣率来反映折扣程度

目标函数的基本原则为，对选中的单品的日收益进行求和，以取得最大值为优化目标，其中求各单品收益的基本计算方式为“销量*定价-进货量*成本”，考虑到商品损耗和打折情况，各单品收益的计算方式修正为“销量*（1-损耗率）*定价+销量*损耗率*折扣率*定价-补货量*成本”（具体公式见下）

$$\max Z_a = \sum_{h=1}^{49} b_h \cdot (S_h + P_h \cdot (1 - loss_h) + S_h \cdot loss_h \cdot discount_h \cdot P_h - amt_h \cdot \cos t_h)$$

$$u_t = \begin{cases} amt_h \geq 2.5 \\ P_h = \cos t_h \cdot w_h \\ amt_h \geq need_h \\ S_h \leq amt_h \\ 27 \leq \sum_{h=1}^{49} b_{\ h} \leq 33 \\ h = 1, 2, 3 \ldots 49 \\ P_h \geq 0 \\ b_h = \{0, 1\} \end{cases} \tag{32}$$

建立约束条件：

根据商超的要求，选择进行补货的单品其订购量需满足最小陈列量 2.5 千克。为体现这一要求，建立约束条件:对于每个单品i，其订购量amt_i需大于等于 2.5。

在定价策略上，本文延续问题二采用的“成本加成定价”方式，即单品销售定价为其批发成本$\cos t_i$加上一定利润率w_i，对于每一个可售单品的定价可以表示为

$$P_h = \cos t_h \cdot w_h \tag{33}$$

此外，还需考虑不同蔬菜品类的市场需求。根据前期分析各品类的最小需求量 Dj，添加约束条件:被选择进行补货的单品在该品类中的总订购量，需要满足对应的 Dj。这可以保证补货计划满足市场对不同品类蔬菜的需求。

在实际情况中，单品的日销量不会超过单品的日补货量：

$$S_h \leq amt_h \tag{34}$$

单品的定价应大于 0：

$$P_h \geq 0 \tag{35}$$

5.4.2.3 使用 BONMIN 求解模型

在求解补货优化模型时，本文采用了 BONMIN(Basic Open-source Nonlinear Mixed INteger programming)求解器（简称“BONMIN”）。BONMIN 是用于解决混合整数非线性规划问题的一种开源工具，它由 IBM 和法国国家计算机科学研究院共同开发，是 COIN-OR 项目的重要组成部分[7]。

BONMIN 实现了基于分支定界法和内点法的算法，可以有效地求解各类凸和非凸 MINLP 问题，处理线性和非线性约束条件。通过算法优化，BONMIN 提供了一个功能强大且灵活的求解平台，可广泛应用于 MINLP 问题。

以下是 BONMIN 优化算法的基本原理：

1) 基于分枝定界法的算法：分枝定界法是一种用于解决整数优化问题的经典方法，它构建了一个解空间的搜索树，并通过递归地划分问题的解空间并去除可忽略的部分，以逐步缩小搜索范围并找到问题的最优解或者可接受的次优解。在 BONMIN 中，分枝定界法被用于处理问题的整数决策变量，它可以有效地处理整数决策变量的离散性，并找到问题的全局最优解或者可接受的次优解。

2) 基于内点法的算法：内点法是一种求解线性规划和非线性规划问题的优化方法，它通过在解空间内部迭代寻找最优解，而不是像单纯形法那样在解空间的边界上进行搜索。在 BONMIN 中，内点法被用于处理问题的非线性约束，它可以有效地处理非线性约束的连续性，并找到满足所有约束条件的最优解。

在本文建立的补货优化模型中，目标函数和约束条件同时包含了非线性成分，且存在整数决策变量，使得不能直接应用线性规划或整数规划算法求解。另外，目标函数中含有商品定价的非线性关系，可能会导致目标函数在解空间中呈现非凸性。BONMIN 求解器可以高效处理这类非凸 MINLP 问题，给出全局最优解或次优解。

本研究通过 Python 的 Pyomo 接口调用 BONMIN 求解器，优化模型代码见附录。经过求解，得到了模型的最优解结果，如下表所示。

表 13 问题三求解结果

索引	西兰花	芜湖青椒(1)	云南生菜	云南生菜(份)	小米椒(份)	竹叶菜	紫茄子(2)	云南油麦菜(份)	螺丝椒	净藕(1)
进货量	13.369	14.493	2.500	33.807	22.519	13.979	11.538	21.584	7.039	6.179
定价	13.599	6.188	11.585	7.137	7.397	4.158	6.737	5.485	15.505	17.023
索引	红薯尖	金针菇(盒)	云南油麦菜	苋菜	娃娃菜	上海青	双孢菇(盒)	奶白菜	螺丝椒(份)	小皱皮(份)
进货量	4.607	16.348	2.500	9.245	10.857	4.018	10.378	6.948	12.469	11.696
定价	5.775	2.538	8.264	3.834	7.583	10.416	5.806	4.991	7.142	3.303
索引	西峡花菇(1)	长线茄	青茄子(1)	菜心	木耳菜	海鲜菇(包)	洪湖藕带	姜蒜小米椒组合装(小份)	菠菜(份)	小青菜(1)

进货量	5.081	4.308	2.924	2.500	5.954	9.603	4.464		7.135	7.458	5.093
定价	27.563	14.100	7.109	6.126	6.656	2.738	21.363		6.064	7.206	6.760

BONMIN 求解器为本文的补货优化问题提供了有效的求解手段，使得商超可以制定出最大化收益的补货方案。

5.5 问题四求解

商超可以考虑采集以下相关数据，这些数据将为商超经营者提供有关市场、消费者和产品的更全面的信息，便于更加准确地预测需求、制定补货计划、调整定价策略，助于提高市场竞争力。

5.5.1 竞争对手数据

竞争对手数据可以提供关于竞争对手的定价策略、促销活动、产品组合等信息。通过分析竞争对手数据，商超可以了解市场上的价格竞争状况，制定合理的定价策略，以保持竞争力。

① 竞争对手的定价策略数据可以反映市场价格水平和价格竞争态势。通过定期跟踪竞争对手的价格，特别是主要品类的价格，商超可以及时了解行业价格变化趋势。这可以帮助商超评估自身价格水平是否合理，避免价格悬殊导致市场份额流失。

② 分析竞争对手的促销活动情况，可以发现市场促销的热点，以及不同促销方式的效果。这为商超制定针对性强、影响力大的促销活动提供依据。若竞争对手增大某品类促销力度，商超也可相应调整该品类的促销策略。

③ 竞争对手的产品组合变化也值得商超关注。如果竞争对手增加或减少某品类商品，可能预示着这类商品的市场需求或供应发生变化。商超可以据此优化自身的品类结构。

5.5.2 市场需求数据

市场需求数据可以提供对蔬菜商品的需求量、消费趋势和偏好的了解。商超可以通过收集市场需求数据来预测和分析市场趋势，合理安排补货和库存管理，以满足消费者需求并避免过剩或缺货情况的发生。

① 了解不同蔬菜品类的市场需求量变化，可以更准确地预测未来需求，制定合理的采购量和库存量，既避免缺货也避免过剩。

② 分析消费趋势变化，能发现消费者需求的转变，比如从传统蔬菜向有机蔬菜的转移，这可以指导商超调整采购结构，增加具有增长潜力的品类。

③ 消费者偏好的数据反映了目标消费群的口味变化，商超可以据此优化产品组合，增加受欢迎的品项，停售不再流行的品项。

④ 不同年龄段、不同收入水平的消费群体对蔬菜的偏好可能有差异，了解这些区别有助于商超更精准地定位和调整产品线。

5.5.3 供应链数据

供应链数据可以提供有关供应商的供货情况、运输时间和成本等信息。商超可以通过分析供应链数据来优化供应链管理，确保蔬菜商品的及时补货和供应，以减少库存风险和满足客户需求。

① 跟踪记录不同供应商的交货周期、到货率等数据，可以评估供应商的稳定性和可靠性，选择更优质的供应商。

② 分析运输时间和成本数据，可以优化运输路线，降低物流成本。例如采用更快或更便宜的运输方式。

③ 根据历史订货和到货数据，可以更准确预测采购量和进货时间，降低缺货风险。

④ 对比不同季节和特殊时间段(如节假日)的供应链情况，可以提前做好应对准备。

5.5.4 节假日涨价与供求情况

节假日涨价与供求情况数据可以提供蔬菜商品在特定节假日或季节的销售情况和价格波动情况。商超可以通过分析这些数据来制定合理的促销策略和定价策略，以应对季节性需求变化和提高销售收益。

① 分析不同节假日蔬菜商品的销量和价格变化情况，可以更准确预测节假日的需求和合理定价。

② 比较不同年份同一节假日的销售数据，可以发现需求变化趋势，预测销量更准确。

③ 综合考量节假日时间的长短、重要程度和需求变化情况，采取有针对性的促销，可以同时提高销量和利润。

④ 对比节假日和平时的价格和促销策略，可以优化节假日的营销方案。

5.5.5 促销效果数据

促销效果数据可以提供关于不同促销活动的销售效果、顾客反馈和市场反应的信息。商超可以通过分析促销效果数据来评估促销活动的效果，确定哪种促销方式对蔬菜商品销售和客户忠诚度影响最大，从而优化促销策略。

① 销售数据分析:通过对促销期间和非促销期间的产品销售量、销售额进行对比，计算出不同促销活动的销售增长率、销售额提升率等效果数据，确定最能刺激销量的促销方式。

② 客户反馈:收集和分析客户对不同促销活动的反馈、评价和建议，包括促销方式的满意度、促销力度的适宜度等，以确定最受客户欢迎的促销方式。

③ 客户粘性分析:评估促销活动前后客户的回头率、购买频次等粘性指标的变化，判断哪些促销更能提高客户忠诚度。

5.5.6 天气数据

天气数据可以提供与蔬菜生长和供应相关的气象信息。天气对蔬菜的种植、产量和品质有重要影响。商超可以通过分析天气数据来预测蔬菜供应的波动，合理安排补货和定价策略，以应对天气变化对供应链的影响。

① 不同地区的天气直接影响当地蔬菜的供应量，需要关注主要产区的天气变化，预测可能的供应波动。

② 极端天气如寒潮、高温、大雨等会严重影响产量和运输，商超需提高应变能力。

③ 根据季节性天气规律，预判蔬菜质量和价格变动趋势，合理安排进货节奏。

④ 针对恶劣天气可能造成的供应紧张，商超可以适当增加库存或提前与其他产区补货。

5.5.7 蔬菜变质速度

蔬菜变质速度数据可以提供蔬菜商品的保鲜期和质量变化情况[8]。商超可以通过了解蔬菜的变质速度来优化库存管理和补货策略，避免库存积压和变质损失，确保顾客购买到新鲜的蔬菜商品。

① 根据不同蔬菜的保质期长度，合理安排库存量，防止过期积压。快变品类适量采购，慢变品类适度储备。

② 优先售卖剩余保质期更短的商品，减少过质亏损。采用先进先出的出库模式。

③ 分析不同保存条件下蔬菜的质量变化情况，优化仓储条件，延长保质期。

六、 模型的评价及推广

6.1 模型的优点

1) 数据预处理

本文在建立和求解各问的模型前，进行了充分的数据预处理步骤，包括：异常值

处理、表格联立，分类汇总、交叉分析等。提高了模型使用数据的质量，为后文模型的效果展示打下了坚实的基础。

2) 可视化统计数据

本文利用了直方图、箱线图、折线图、散点图等多种可视化手段去描述数据的分

布规律，能够清晰地了解到数据的相关特征。

3) 模型检验

本文对所建立的模型，均进行了严格的模型验证，包括：残差分析、预测值与真实值比对、相关系数计算和 F 显著性检验等。更加说明了本文所建立模型较强的合理性和良好的效果。

6.2 模型的不足

程序运行时间较长。本文在第三问中建立了非线性的混合整数规划，此模型对计算机要求较高，需要提高计算机配置，采用启发式算法才能快速求解。

6.3 模型的推广

本文中构建了加入 0-1 变量的混合整数规划模型，此模型方法灵活，目前已成功应用于求解生产进度问题、旅行推销员问题、工厂选址问题、背包问题及分配问题等，有较强的推广性。

七、 参考文献

[1] Dutilleul P， Stockwell J D， Frigon D， et al. The Mantel test versus Pearson's correlation analysis: Assessment of the differences for biological and environmental studies[J]. Journal of agricultural， biological， and environmental statistics， 2000: 131-150.

[2] Chang， Lina， Wang， Yingli， Wang， Yao. Classification and prediction of urban air quality classes based on K-means clustering and Bayesian discriminant[J]. Journal of Taiyuan Normal College (Natural Science Edition)，2021，20(02):41-46.

[3] Cleveland R B， Cleveland W S， McRae J E， et al. STL: A seasonal-trend decomposition[J]. J. Off. Stat， 1990， 6(1): 3-73.

[4] 葛娜，孙连英，赵平等.基于 ARIMA 时间序列模型的销售量预测分析[J].北京联合大学学报，2018，32(04):27-33.DOI:10.16255/j.cnki.ldxbz.2018.04.006.

[5] Fracas， P.， Camarda， K. & Zondervan， E. (2023). Shaping the future energy markets with hybrid multimicrogrids by sequential least squares programming. Physical Sciences Reviews， 8(1)， 121-156. https://doi.org/10.1515/psr-2020-0050.

[6] 胡应钢，郭翔，赵海燕等.基于粒子群优化 GRU-RNN 组合模型的云计算资源负载预测[J].内蒙古民族大学学报(自然科学版)，2023，38(04):315-321.DOI:10.14045/j.cnki.15-1220.2023.04.005.

[7] Ahmed EM， Rakocˇevic´ S， C´ alasan M， Ali ZM， Hasanien HM， Turky RA， et al. BONMIN solver-based coordination of distributed FACTS compensators and distributed generation units in modern distribution networks. Ain Shams Eng J 2022;13:. doi: https://doi.org/10.1016/j.asej.2021.101664101664.

[8] 卢亚杰. 我国超市优质生鲜蔬菜动态定价问题研究[D].北京交通大学，2010.

附录

支撑材料列表：

第二问各品类_定价_销量_频数/第二问受频数影响的数据处理.py

第二问规划求解/第二问规划求解.py

ARIMA.ipynb

箱线图(品类).ipynb

销量.ipynb

未来 7 天成本预测.xlsx

回归方程展示图.ipynb

第三问回归处理.ipynb

第三问 gru 算法.ipynb

第三问 gru 预测结果.xlsx

按月销售量汇总.ipynb

单品合并.ipynb

50 个单品对应线性结果.xlsx

平均损耗率（按类）.xlsx

arima 预测结果.xlsx

最优化系数表.xlsx

按月汇总（商品序列）.xlsx

单品数据总表.xlsx

单品销售量.xlsx

第三问 gru 预测结果.xlsx

第三问预测销量和成本_三月.xlsx

第一问聚类分析.xlsx

品类数据总表（不含公式）.xlsx

平均损耗率（按类）.xlsx

附录 1
介绍：对附件 2.xlsx 的数据预处理程序 按月销售量汇总.ipynb
import pandas as pd import os from openpyxl import load_workbook df = pd.read_excel('附件 2.xlsx') df['销售日期'] = pd.to_datetime(df['销售日期']) df['年'] = df['销售日期'].dt.year

```
df['月'] = df['销售日期'].dt.month
df_grouped = df.groupby(['单品编码', '年', '月'])['销量(千克)'].sum().reset_index()
df_grouped = df_grouped.sort_values(['年', '月'])
df_grouped = df_grouped.pivot_table(index='单品编码', columns=['年', '月'], values='销量(千克)', fill_value=0)
df_grouped.columns = [f'{year}-{month}' for year, month in df_grouped.columns]
df_grouped = df_grouped.loc[:, '2020-7':'2023-6']
with pd.ExcelWriter('按月汇总.xlsx') as writer:
    df_grouped.to_excel(writer, startrow=1)
```

附录 2

介绍： 对附件 2.xlsx 按月份为索引切割文件 切割文件.ipynb

```
import pandas as pd
import os
from openpyxl import load_workbook

# Load the excel file
df = pd.read_excel('附件 2.xlsx')

# Convert the first column to datetime
df['销售日期'] = pd.to_datetime(df['销售日期'])

# Group by year and month
grouped = df.groupby([df['销售日期'].dt.year, df['销售日期'].dt.month])

# Loop through each group
for (year, month), group in grouped:
    # Create a directory for the month if it doesn't exist
    directory = f'{year}年{month}月'
    if not os.path.exists(directory):
        os.makedirs(directory)

    # Save the group to a new excel file in the directory
    group.to_excel(f'{directory}/{year}年{month}月.xlsx', index=False)
```

附录 3

介绍：使用 matplotlib 绘制品类箱线图　箱线图(品类).ipynb

```
import pandas as pd

# Load spreadsheet
```

```
xl = pd.ExcelFile('按月汇总（类）.xlsx')

# Load a sheet into a DataFrame by name
    df1 = xl.parse('品类按月销售量（转置）')
    from pylab import mpl
    mpl.rcParams['font.sans-serif'] = ['Microsoft YaHei'] # 指定默认字体：解决 plot 不能显示中文问题
    mpl.rcParams['axes.unicode_minus'] = False # 解决保存图像是负号'-'显示为方块的问题
    import matplotlib.pyplot as plt

    # Select the columns for the boxplot
    boxplot_data = df1[['花叶类', '花菜类', '水生根茎类', '茄类', '辣椒类', '食用菌']]

    # Create a figure instance
    fig = plt.figure(figsize =(10, 7))

    # Create an axes instance
    ax = fig.add_subplot(111)

    # Create the boxplot
    bp = ax.boxplot(boxplot_data.values)

    # Change the labels
    ax.set_xticklabels(['花叶类', '花菜类', '水生根茎类', '茄类', '辣椒类', '食用菌'])

    # Set the y-axis label
    ax.set_ylabel('销量')
    ax.set_xlabel('品类名称')
    # Show the plot
    plt.show()
```

附录 4

介绍：使用 matplotlib 绘制部分单品箱线图　箱型图(商品).ipynb

```
import pandas as pd

df = pd.read_excel('按月汇总（商品序列）.xlsx')
import seaborn as sns
    print(df)
    import matplotlib.font_manager as fm

    from pylab import mpl
    mpl.rcParams['font.sans-serif'] = ['Microsoft YaHei'] # 指定默认字体：解决 plot 不
```

```
能显示中文问题
    mpl.rcParams['axes.unicode_minus'] = False # 解决保存图像是负号'-'显示为方块的问题
    import seaborn as sns
    # df.iloc[:, 1:] = (df.iloc[:, 1:] - df.iloc[:, 1:].mean()) / df.iloc[:, 1:].std()
    # print(df)
    import seaborn as sns
    import matplotlib.pyplot as plt

    n = 10
    largest_columns = df.iloc[:, 1:].sum().nlargest(n).index

    df_largest = df[largest_columns]

    plt.figure(figsize=(8, n*0.5))
    sns.boxplot(data=df_largest, orient='horizontal', linewidth=2, width=0.5)
    plt.xlabel('月均销量')
    plt.ylabel('单品名称')

    plt.title('单品箱线图')
    plt.show()
```

附录 5

介绍：使用 matplotlib 进行直观的回归方程展示　回归方程展示图.ipynb

```
    from statsmodels.tsa.arima.model import ARIMA
    # ARIMA(data, order=(p, d, q))
    model = ARIMA(diff_mte, order=(6,2,4))
    result = model.fit()
    result.summary()
    import pandas as pd

    name = "食用菌"
    file_path = f"分频数各品类回归数据/{name}summary.xlsx"

    data = pd.read_excel(file_path, sheet_name='Sheet1')
    print(data)
    import matplotlib.pyplot as plt
    file_path = f"分频数各品类回归数据/x,y 值.xlsx"

    data2 = pd.read_excel(file_path, sheet_name='Sheet1')
    print(data2)
    intercept = data2.loc[data2['分类名称'] == name, '截距'].values[0]
    print(intercept)
```

```
slope = data2.loc[data2['分类名称'] == name, '斜率'].values[0]
import numpy as np

x = np.arange(data['单日平均定价(元)'].min(), data['单日平均定价(元)'].max(), 0.01)
y=intercept+x*slope
plt.scatter(data['单日平均定价(元)'], data['当日销量(kg)'], s=data['频数'], label='频数权重点')
plt.plot(x, y, color='red', label='线性回归预测值')
plt.xlabel('单日平均定价(元)')
plt.ylabel('当日销量(kg)')
plt.title(name+'回归方程拟合图')
plt.legend()
plt.show()
```

附录 6

介绍：使用的 ARIMA 模型核心代码　ARIMA.ipynb

```
##完整代码见文件 ARIMA.ipynb，篇幅有限，只展示核心代码，省去了繁琐的模型检验环节
from statsmodels.tsa.arima.model import ARIMA
# ARIMA(data, order=(p, d, q))
model = ARIMA(diff_mte, order=(6,2,4))
result = model.fit()
result.summary()
```

附录 7

介绍：用 ARIMA 模型预测数据集　ARIMA.ipynb

```
# 预测出来的数据也为一阶差分
# predict(起始时间，终止时间)
predict = result.predict('2020-07-01','2023-06-30')
plt.figure(figsize=(12, 8))
plt.plot(diff_mte)
plt.plot(predict)
```

附录 8

介绍：SLSQP 解决第二问规划问题　cys 代码文件/第二问规划求解/第二问规划求解.py

```
from scipy.optimize import minimize
import pandas as pd

def optimize(b, k, loss, discount, cost, X_low, Y_low, Y_high):
    # 目标函数
    def objective_function(variables):
        X, Y = variables
        Z = -(b-k*Y)*(1-loss)*Y - (b-k*Y)*loss*discount*Y + X*cost
        return Z

    # 约束条件
    constraints = (
        {'type': 'ineq', 'fun': lambda variables: variables[0] - X_low},  # X>=X_low
        {'type': 'ineq', 'fun': lambda variables: variables[1] - Y_low},  # Y>=Y_low
        {'type': 'ineq', 'fun': lambda variables: -variables[1] + Y_high},  # Y<=Y_high
        {'type': 'ineq', 'fun': lambda variables: variables[0] - (b-k*variables[1])}  # b-
k*Y<=X
    )

    # 初始猜测
    initial_guess = [10, (Y_low+Y_high)/2]

    # 求解
    solution = minimize(objective_function, initial_guess, constraints=constraints)

    # 输出结果
    print(f"The optimal value of X is: {solution.x[0]}")
    print(f"The optimal value of Y is: {solution.x[1]}")

# 使用不同参数运行优化
print('-'*40)


# 读取 Excel 文件
df = pd.read_excel('最优化系数表.xlsx', sheet_name='食用菌')

# 遍历每一行
for index, row in df.iterrows():
    b = row['b']
    k = row['k']
    loss = row['loss']
    discount = row['discount']
    cost = row['cost']
    X_low = row['X_low']
    Y_low = row['Y_low']
```

```
    Y_high = row['Y_high']

    # 调用优化函数
    optimize(b, k, loss, discount, cost, X_low, Y_low, Y_high)
    print('-' * 40)
```

附录 9

介绍：gru 算法预测 7 月 1 日单品销量与成本　第三问 gru 算法.ipynb

```
    inputname='成本' #可填销量/成本 表示预测 7 月 1 日的哪个值# predict(起始时间，终止时间)
    import pandas as pd
    from keras.models import Sequential
    from keras.layers import GRU, Dense
    from sklearn.preprocessing import MinMaxScaler
    import numpy as np

    # 加载数据
    df = pd.read_excel('第三问预测销量和成本_三月.xlsx',sheet_name=inputname)

    # 移除第一列
    df = df.iloc[:, 1:]

    # 初始化预测结果
    predictions = []

    # 对每一列进行预测
    for i in range(df.shape[1]):
      # 获取列数据
      data = df.iloc[:, i].values
      # 删除 data 中的 0 值  !!! 如果是预测销量，我们组认为不需要去零处理，而成本需要去 0 处理
      data = data[data != 0]
      # 数据归一化
      scaler = MinMaxScaler(feature_range=(0, 1))
      data = scaler.fit_transform(data.reshape(-1, 1))

      # 创建数据集
      X, y = [], []
      for i in range(len(data)-1):
        X.append(data[i])
        y.append(data[i+1])
      X, y = np.array(X), np.array(y)
```

```
        # 重塑输入为 [samples, timesteps, features]
        X = X.reshape((X.shape[0], 1, X.shape[1]))

        # 定义 GRU 模型
        model = Sequential()
        model.add(GRU(50, activation='relu', input_shape=(1, 1)))
        model.add(Dense(1))
        model.compile(optimizer='adam', loss='mse')

        # 训练模型
        model.fit(X, y, epochs=100, verbose=0)

        # 对下一步进行预测
        x_input = np.array(data[-1]).reshape((1, 1, 1))
        yhat = model.predict(x_input, verbose=0)

        # 反归一化预测结果
        yhat = scaler.inverse_transform(yhat)

        # 将预测结果添加到预测列表
        predictions.append(yhat[0,0])

    # 将预测结果添加到数据框最后一行
    df.loc[df.shape[0]] = predictions
```

附录 9

介绍：用 ARIMA 模型预测数据集　ARIMA.ipynb

```
        # 预测出来的数据也为一阶差分
        # predict(起始时间，终止时间)
        predict = result.predict('2020-07-01','2023-06-30')
        plt.figure(figsize=(12, 8))
        plt.plot(diff_mte)
        plt.plot(predict)
```

附录 10

介绍：求解第三问混合规划代码　cys 代码文件/第三问求解混合规划.py

```
    from scipy.optimize import minimize
    import pandas as pd

    def optimize(b, k, loss, discount, cost, X_low, Y_low, Y_high):
      # 目标函数
      def objective_function(variables):
```

```
        X, Y = variables
        Z = -(b-k*Y)*(1-loss)*Y - (b-k*Y)*loss*discount*Y + X*cost
        return Z

    # 约束条件
    constraints = (
        {'type': 'ineq', 'fun': lambda variables: variables[0] - X_low},  # X>=X_low
        {'type': 'ineq', 'fun': lambda variables: variables[1] - Y_low},  # Y>=Y_low
        {'type': 'ineq', 'fun': lambda variables: -variables[1] + Y_high},  # Y<=Y_high
        {'type': 'ineq', 'fun': lambda variables: variables[0] - (b-k*variables[1])}  # b-k*Y<=X
    )

    # 初始猜测
    initial_guess = [10, (Y_low+Y_high)/2]

    # 求解
    solution = minimize(objective_function, initial_guess, constraints=constraints)

    # 输出结果
    print(f"The optimal value of X is: {solution.x[0]}")
    print(f"The optimal value of Y is: {solution.x[1]}")

# 使用不同参数运行优化
print('-'*40)


# 读取 Excel 文件
df = pd.read_excel('最优化系数表.xlsx', sheet_name='食用菌')

# 遍历每一行
for index, row in df.iterrows():
    b = row['b']
    k = row['k']
    loss = row['loss']
    discount = row['discount']
    cost = row['cost']
    X_low = row['X_low']
    Y_low = row['Y_low']
    Y_high = row['Y_high']

    # 调用优化函数
    optimize(b, k, loss, discount, cost, X_low, Y_low, Y_high)
        print('-' * 40)
```