

蔬菜定价与补货的预测与规划模型

摘要

针对问题一，对于第一问，先求出变异系数、偏度、峰度等描述统计量，同时使用**箱线图**刻画分布。其次利用**时间序列**的分解挖掘销量分布的趋势性和季节性讨论时间序列的周期性性质，各品类的周期全部为 7。最后对序列进行**多项式拟合**。对于第二问，先用**斯皮尔曼相关系数**初步探究数据关联性，再采用**灰色关联分析**进一步分析数据关联性，品类中花菜类和花叶类的关联系数最高，为 0.9，单品中云南生菜与红椒 (1) 关联系数最高，为 0.8。最后，本文使用 FP-tree 算法挖掘商品之间的关联规则，置信度最高 0.92，为 (净藕 (1), 奶白菜 (份), 菜心 (份), 金针菇 (盒))→(芜湖青椒 (1))。

针对问题二，对于第一问，本文采用**集成学习**和**梯度增强**改进决策树得到 **XGBoost** 模型，结果表示该模型预测结果 R^2 得分为 0.81，MAE 为 15.8kg，效果较好。对于第二问，基于 ACF 和 PACF 图，本文选择 **AR 自回归模型**对批发价进行预测。其次基于 XGBoost 对未来一周销量关于定价关系的预测，以利润为优化目标建立**添加高斯噪声的非线性规划模型**，使用**遗传 (GA)** 算法求解。最终结果表明各品类定价在 4-25 之间，补货量在 23-200 之间，打折率在 0.75-1 之间，一周的日利润在 1600-1800 元之间，相比优化前有显著提升。最后建立**时间衰减折扣模型**作为附加策略。

针对问题三，本文先使用**相关性聚类编码**单品名和所属大类名，并增加其作为特征输入 XGBoost 以提升 R^2 得分。其次在问题二的基础上增加示性变量为决策变量，并且增加可售单品总数，最小陈列量和满足需求的约束，同样使用遗传算法求解该非线性规划模型。结果表明单品定价在 1.5-9 之间，补货量在 2-25 之间，打折率在 0.75-1 之间，可售单品总数为 29，最大日利润为 1338 元。

针对问题四，本文列举了季节/天气性数据，顾客流量数据，重大意外事件，蔬菜供应链数据，附近商超的销售政策流量数据，节假日数据，存储和运输成本以及损耗风险数据，新鲜度下降数据，消费者数据，并且分析了这些数据对于本问题的益处并提出建议。

【关键词】 灰色关联分析 FP-tree XGBoost AR 自回归模型 高斯噪声 多目标非线性规划 GA

1 问题重述

在现代生鲜商超中，蔬菜的供应和销售是一个复杂而又必要的业务环节。由于蔬菜短暂的保鲜期和品相的易变性，如果商超不能在当天销售出，蔬菜可能在隔日就失去了销售价值。这就要求商超在每日的凌晨，根据前一天的销售情况和当前市场需求来进行补货决策。但是，由于蔬菜品种众多、产地不同，以及供货时间的特殊性，使得这一补货决策充满了挑战。因此，对于商超来说，如何准确地分析市场需求，合理地制定补货和定价策略，对于确保销售收益、减少浪费和满足消费者需求具有重要意义。因此，凭借附件中提供的关于六种蔬菜品类的商品信息，销售流水明细，批发价格数据和损耗率数据，我们将深入研究以下几个问题：

- (1) **问题一** 分析并探究蔬菜类商品不同品类和单品销售量的分布规律及其之间的相互关系；
- (2) **问题二** 考虑按品类进行补货计划，探索蔬菜品类的销售总量与成本加成定价的关系；并为未来一周提供日补货总量和定价策略建议。
- (3) **问题三** 基于销售空间的限制，提出一个精细化的蔬菜单品补货计划和定价策略；以满足市场需求并最大化商家收益。
- (4) **问题四** 分析商超为更好地制定蔬菜商品的补货和定价决策，还需采集的其他相关数据。

2 模型假设

- (1) 评估每一品类单日成本加成定价时，按该品类下所有单品售价关于销量的加权平均估计；
- (2) 假设商超一分钟内的所有订单来自于同一个顾客；
- (3) 假设优化求解模型中不考虑单天内蔬菜品相的损坏；
- (4) 假设销量的预测值是当天市场对该单品的需求量。

3 符号说明

符号	含义	单位
α_i	第 i 种单品/品类蔬菜定价	元/kg
β_i	第 i 种单品/品类蔬菜打折率	
k_i	第 i 种单品/品类蔬菜补货量	kg
W	总收益	元

表 1: 主要符号说明，其余符号详见文中说明

4 问题分析

4.1 问题 1 分析

该问题旨在探究蔬菜类商品不同品类或不同单品之间的关联关系，可以分为两部分：

- (1) 对于各品类和单品的销量-时间分布进行统计学性质的刻画；
- (2) 在各品类和各单品之间进行分析，寻找潜在的相关性。

第 (1) 部分属于统计量化问题。对于销量-时间的序列型信息，首先给出均值、方差等描述性统计指标，同时利用时间序列的分解挖掘销量分布的趋势性和季节性。考虑到蔬菜销量对于季节的高度敏感性，我们基于小波分析进一步讨论时间序列的周期性性质。最后对序列进行拟合，得到显式的函数关系。

第 (2) 部分涉及到关联性分析和数据挖掘。在研究各品类和各单品之间的相关关系时，先使用 Spearman 相关系数初步探究序列数据中线性的关联以及更一般的关联性。在此基础之上，采用灰色关联分析进一步分析数据间的关联性。考虑到问题的背景，蔬菜类商品的销售往往伴随着捆绑购买的现象，也可能会出现互补的现象。依据这个观察我们使用数据挖掘的手段，基于 FP-tree 算法挖掘商品之间的关联规则。

4.2 问题 2 分析

该问题旨在寻找使得商超收益最大的日补货总量和定价策略。这个问题可以分为以下三个部分解决：

- (1) 分析得到目标日期附近各蔬菜品类的销售总量与成本加成定价的关系；
- (2) 基于蔬菜类商品的历史批发价格对未来 7 天的批发价格进行预测；
- (3) 根据 (1) 中的函数关系和 (2) 中的预测值，建立以商超收益为目标函数的优化模型，制定日间和日内的最佳定价策略，给出 7 日内最合理的补货量分布。

第 (1) 部分属于回归问题。考虑到数据的时间特征不可忽略，我们将时间进行编码，结合该天的成本加成定价和打折率作为输入，以该天的销售总量作为标签建立 XGBoost 机器学习模型，得到销售总量与成本加成定价的映射关系。

第 (2) 部分是时间序列的预测问题。由于问题背景说明了商家在做补货决策的时候是不确切知道具体单品和进货价格的，这里先用附录 3 的数据关于销售量加权得到各品类的平均批发价格的时序数据，基于 ACF 图和 PACF 图对数据的时序特征的特征，选用合适的时间序列模型对于每个品类的平均批发价格进行预测。

第 (3) 部分是优化类问题。首先建立商超收益的表达式，并将其作为优化的目标函数。考虑到需求侧的不稳定性，我们在目标函数中加入了高斯噪声以增强模型的稳健性。最后根据实际情况添加约束条件，使用遗传算法进行求解。

4.3 问题 3 分析

该问题在问题二的基础上增加了求可售单品总数和最小陈列量的约束，此外考察的对象变为了单品。首先，要在问题二的 XGBoost 模型之上针对特定单品训练新的模型。此外，需要着眼于特定的单品的历史批发价格对未来 1 天的批发价格进行预测。最后对问题二的目标函数进行微调，引入示性变量和新的约束利用遗传算法进行求解。

4.4 问题 4 分析

该问题在前三个问题的基础之上，考虑题目背景未给出的诸多因素如节日、重大事件（疫情等）、竞争因素、供应数据等等，并且给出这些数据对解决上述问题的重要性评估。

5 数据概览

本文处理的问题为数据分析问题，本节旨在总览数据，对于本次任务处理的数据进行可视化，了解数据的大致性质从而为后续建立合理的模型建立可靠的先验。

5.1 商品销量情况概览

汇总数据，得到数据的基本信息如下表：

表 2: 数据的基本信息

品类/单品	总数	销量最大值 (kg)	销量最小值 (kg)	销量偏度
品类	6	198,520.978	22,431.782	24.331
单品	246	28,164.331	0.415	1.857

品类和单品的销量分布均为正偏度，证明存在某些时间段，某些品类或单品的销售量异常高，单品的极差很大也说明了这一点。下表展示了一些边缘化商品：

表 3: 部分边缘化商品

单品编码	单品名称	批发价记录天数	是否在最近一月出现
102900005115748	白菜苔	97	否
102900005115977	豌豆尖	45	否
102900011026618	艾蒿	5	是

所有在三年内批发价的记录次数小于 100，除了艾蒿外均未在近一月内出现。一共有 130 个单品满足这一点。这些单品数据点较少，不适于后续的时序分析，故在后续预测批发价等工作中忽略这些单品。

图 1 显示了蔬菜品类三年来销量的分布和销量最高的 10 个单品分布：

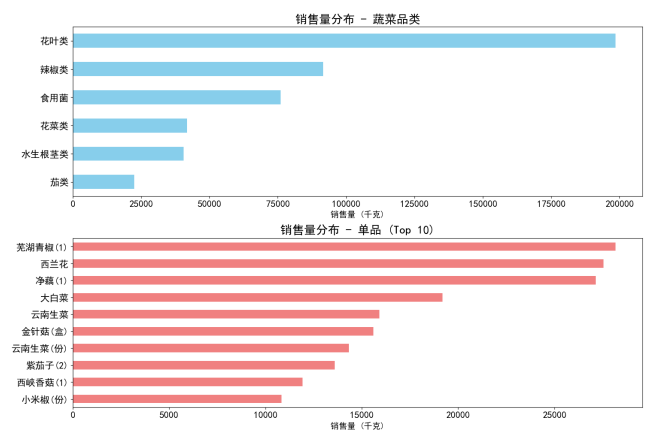


图 1: 蔬菜品类销量的分布与销量

可以看出，花叶类蔬菜销量最好，总销售量约为茄类的 9-10 倍。单品方面，芜湖青椒 (1)、西兰花和净藕 (1) 的销量最大，销量约为排名第 10 的小米椒 (份) 的两倍。上图也显示了对于单品，数据分布的右尾部分比左尾部分长，证明有不少潜在的“边缘化”的商品，可能对后续的分析造成影响。

5.2 打折规律

折扣是销售的重要的一部分。在本文的背景里，打折作为促销手段，必然有的单品“易于”成为促销产品，而有的产品无需打折即可卖完。在这里定义打折比例为为该单品打折的总销量除以该单品的总销量。我们从近一个月有流水记录的商品中进行统计，得到部分结果如下：

表 4: 部分单品打折比例

单品编码	单品名称	打折比例
102900011036686	木耳菜 (份)	0.4074
102900011030929	鲜木耳 (份)	0.4062
102900011032732	高瓜 (2)	0.3148
.....		
102900011018132	野生粉藕	0
102900011007969	红莲藕带	0

数据显示只有艾蒿、野生粉藕和红莲藕带三种单品从未打折销售，而单品木耳菜 (份) 和鲜木耳 (份) 的打折比例高达 40% 。

下面以近一个星期各品类的打折销售情况为例：

表 5: 各品类的打折销售情况

分类名称	打折的销量	总销量	打折占比
水生根茎类	41.144	116.702	35.3%
花叶类	122.898	873.928	14.1%
花菜类	5.102	113.107	4.5%
茄类	0.000	113.105	0.0%
辣椒类	33.109	543.083	6.1%
食用菌	63.000	309.369	20.4%

表 5 显示，水生根茎类打折的比例最大，花叶类和食用菌也有较大的打折比例，而茄类未曾打过折。证明不同品类的销售策略不一样，这或许与各品类的新鲜度随时间的变化特征有关。本文的问题中，“打折”的动机有两种：一种是由于进货时原本的损失导致的打折，这往往出现在较早的时间点；另一种是由于菜品新鲜度降低而催生的促销策略，后者往往出现较晚。可视化展示如图 2:

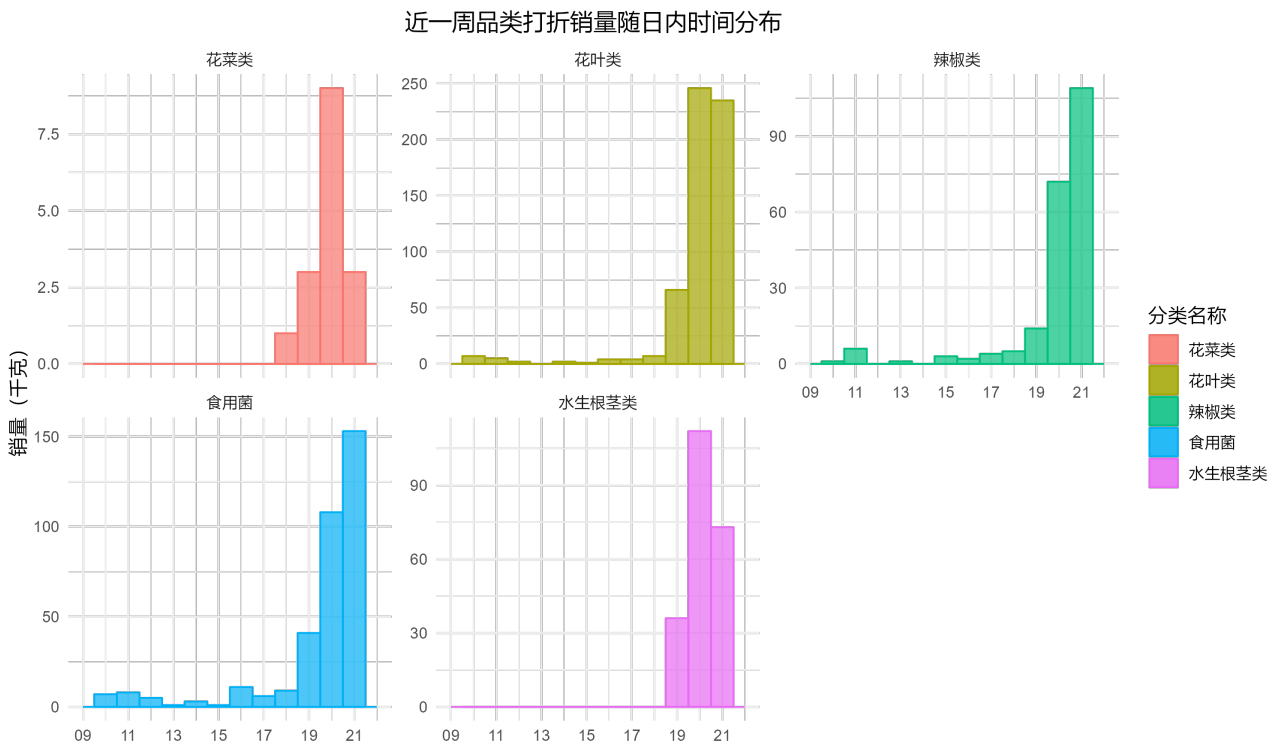


图 2: 各品类的日内打折分布

可以看出，食用菌、花叶类和辣椒类在早上用过少量的打折，而其他类型在早上并不会直接开始促销。此外，绝大多数打折订单均出现在晚上，且花菜类和水生根茎类仅在晚上开始打折。

6 模型建立与求解

6.1 问题 1 数据评价与分析

本节先对各品类销量概况进行可视化，接着从一些统计量量化各个品类和单品的销量分布情况。

统计量量化

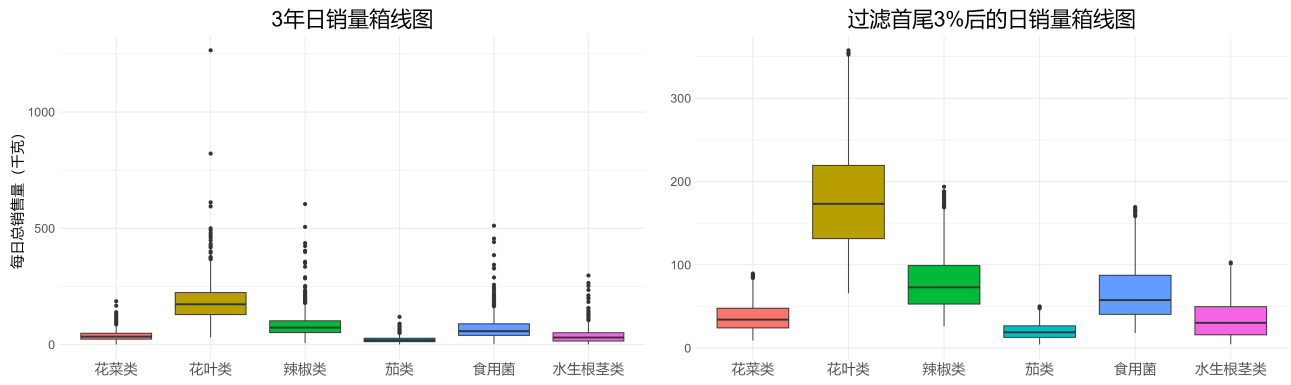


图 3: 三年各品类日销量

绘制 3 年来各品类日销售量箱线图如图 3 所示，可见花叶类有较多分布极端的点，去掉首尾 3% 的数据后得到下图。需求量最大的是花叶类，食用菌和辣椒类次之，其余类需求较少，波动范围较为均匀，都存在少量极端的样本点。

(1) 变异系数

变异系数是相对标准差的度量，用于描述数据的离散程度，特别是当需要比较两个或多个数据的离散程度时有效。样本的变异系数计算方法为：

$$CV = \frac{\mu}{\sigma} \quad (1)$$

其中， μ 为样本均值， σ 为样本方差。变异系数可以用于比较不同单位或量纲的数据集的变异性。

(2) 偏度

偏度描述了概率分布的偏斜（或不对称）程度。它可以是正值，负值或零。正偏度意味着数据的右尾部分较长，大部分值位于平均值的左侧。反之同理。样本的偏度计算方法为：

$$g_1 = \frac{\mu_3}{S^{\frac{3}{2}}} \quad (2)$$

其中， μ_3 是三阶中心矩， S 是样本方差。

(3) 峰度

峰度描述了概率分布的尖度。与正态分布相比，它可以用于描述数据尾部的重要性和风险。峰度的计算公式为：

$$g_2 = \frac{\mu_4}{S^2} - 3$$

(3)

其中， μ_4 是四阶中心矩。正峰度意味着与正态分布相比，数据有更重的尾部和更尖的峰；负峰度则代表与正态分布相比，数据有更轻的尾部和更扁平的峰。

对于 6 个品类，其各个统计量数据如下：

表 6: 品类的统计量

分类名称	最大值	最小值	平均值	标准差	变异系数	偏度
水生根茎类	296.8	1.0	37.4	31.357	0.838	2.508
花叶类	1265.5	31.3	183.0	86.199	0.471	2.901
花菜类	186.2	0.6	38.5	22.675	0.589	1.528
茄类	118.9	0.3	21.4	13.159	0.616	1.738
辣椒类	604.2	6.1	84.4	53.436	0.633	3.146
食用菌	511.1	3.0	70.1	48.490	0.691	2.996

花叶类商品在这些品类中拥有最大的销量范围和最高的平均销量，这意味着它包含一些较受欢迎的单品或特定时段的高销售额。相对地，茄类在所有品类中的销量相对稳定且平均销量较低。

所有品类的销量数据都呈现出右偏分布，这表明少数日子的销量特别高。尤其是花叶类和辣椒类，他们不仅有更高的峰度，表明其销量分布在平均水平附近较为尖锐，而且很可能存在更多的异常高销售日。

部分单品的统计量数据如下：

表 7: 部分单品的统计量

分类名称	最大值	最小值	平均值	标准差	变异系数	偏度
七彩椒 (1)	16.1	0.1	0.93	1.408	1.519	7.059
七彩椒 (2)	5.8	0.1	1.4	1.055	0.736	1.290
七彩椒 (份)	4.0	1	1.8	0.919	0.511	1.546
.....						
黄白菜 (2)	187.7	0.2	8.8	10.469	1.185	6.776
黄花菜	0.7	0.1	0.30	0.230	0.774	0.881
黑油菜	29.6	0.3	3.7	5.571	1.487	4.146

首先，从销量范围来看，娃娃菜是最受欢迎的单品，其平均销量高达 32.95，最高日销量高达 407kg。这不仅意味着娃娃菜在某些特定时段的销售额特别高，而且在整个三年期间都受到消费者的喜爱。相对地，白蒿的平均销量为 0.66，最低值是-1，这表明它是一个较为平庸商品。七彩椒 (1) 也表现出了显著的销量特征。其偏度为 7.06，仅次于娃娃菜，这表明它在少数日子的销量也表现得相当出色。这或许表明，在某些促销或节假日时段，消费者可能对某些商品有更强烈的购买意愿。整体上，所有单品的销量数据都呈现出

明显的右偏分布，平均偏度为 1.84。这种偏度分布说明我们的销量数据中，高销量的日子相对较少，但销售表现特别好。此外，销量分布的峰度平均为 9.50，远大于正态分布的峰度，这意味着销量分布在平均水平附近较为尖锐，可能存在更多的异常高销售日。

(4) **周期性**下面就花叶类在过去一年的每天销量变化做初步的观察。对此时间序列进行 ADF (Augmented Dickey-Fuller) 检验，原假设是时间序列具有单位根，即它是非平稳的。在这里 ADF 检验的 p 值为 0.0208，因此，我们拒绝原假设，认为该时间序列是平稳的，进而分解如下：

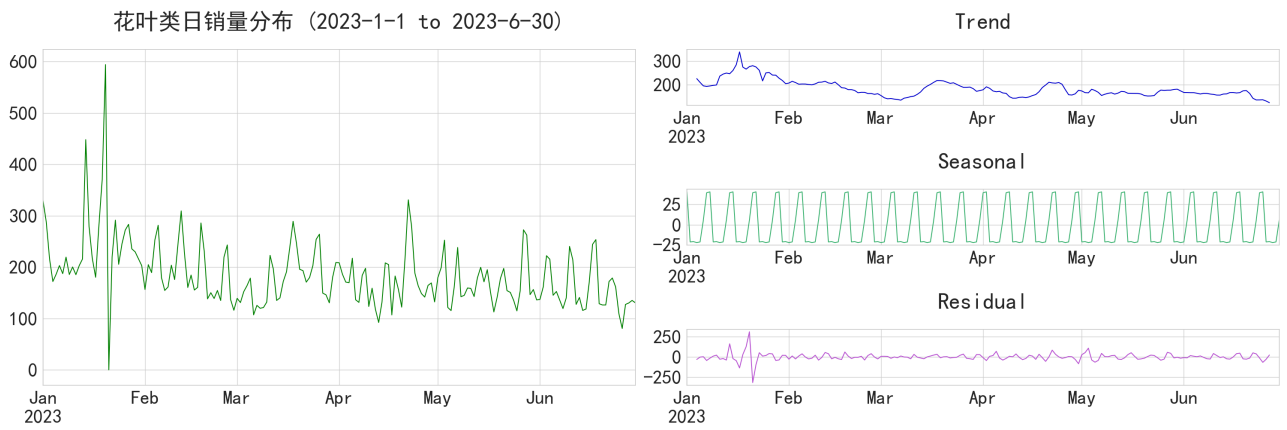


图 4: 花叶类近半年日销量分布和时间序列分解

通过查找季节性组件的自相关函数 (ACF) 的峰值来确定季节性组件的周期大小，经编程得到每一个品类季节项的周期均为 7。可见销量分布的涨落以一星期为周期震荡变化，这与生活常识相符。由于对于单品，有可能有大量天数数据缺失，因此在品类的层面考察周期性更为合理，这里便不考虑单品的周期性性质。

关联性分析

本小结着眼于品类和单品内部的相关性，我们将从斯皮尔曼相关系数出发，再进一步使用灰色关联分析深入探究。

(1) 斯皮尔曼相关系数

斯皮尔曼相关系数测量两个变量的秩之间的单调关系。相比于皮尔逊相关系数，它并不假定样本来源于近似正态的分布，它测量的是两个变量之间的单调关系，且不仅仅是线性。其计算公式如下：

$$\rho = 1 - \frac{6 \sum_{i=1}^n d_i^2}{n(n^2 - 1)} \tag{4}$$

其中， d_i 是两个变量的秩之间的差异， n 是数据点的数量。对于 6 个品类，得到结果如下：

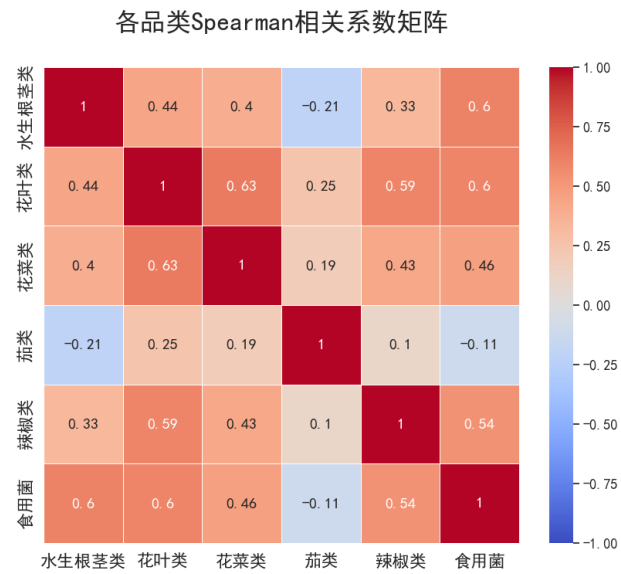


图 5: 品类之间斯皮尔曼相关系数矩阵

可以观察到，茄类与其他所有类相关性都不是很强，而花叶类、水生根茎类与除茄类以外的其他类均有正相关关系。

在分析单品之间的性质时，注意到单品有货的天数有高有低，因此在计算时筛选出了至少 100 天有货的产品，且在计算斯皮尔曼相关系数是，若该天两个单品销量均为 0，则去掉这一天的数据。下面给出单品之间斯皮尔曼相关系数的一些典型样本：

表 8: 部分单品打折比例

单品名称 1	单品名称 2	Spearman 相关系数
云南油麦菜 (份)	云南生菜 (份)	0.683762
金针菇 (1)	青线椒	0.536175
杏鲍菇 (1)	姬菇 (1)	0.500968
.....		
云南生菜	云南生菜 (份)	-0.745892
杏鲍菇 (2)	海鲜菇 (袋)(4)	-0.857227
青红杭椒组合装 (份)	泡泡椒 (精品)	-0.857838

可以看出，顾客买了组合装就不会买类似的商品，同理，对于同一商品的不同包装形式，顾客会选择其一购买，导致出现了强负相关的“互补型”销量趋势，这也是符合生活经验直观的。也可以观察到，类似的商品如油麦菜、生菜和杏鲍菇、姬菇有着相似的售卖趋势。

(2) 灰色关联分析

灰色关联分析（Grey Relational Analysis, GRA）是灰色系统理论的一部分，用于确定不同因素之间的关联度。在本文中它可以评估时间序列数据之间的模式相似度^[4]。灰色关联分析是一种对影响目标序列的

相关因素进行定量分析的模型，它通过对比各个序列对目标序列的数据增长或减少的相似或相异程度来判断各个因素对目标序列的影响结果。相比于前文的方法，灰色关联分析法对非线性关系更敏感，适用于小样本且不受异常值干扰。数据处理与计算步骤如下：

Step 1 数据标准化

为了消除数据的量纲和数值差异，首先需要进行标准化处理。因此参与比较的数据均为 n 维向量，设为 $x_i, i = 0, 1, \dots, r$ ， n 为 3 年的天数，为参与比较的序列数。标准化的目的在于消除量纲影响，维持原有分布。这里采用除以均值的办法，即

$$x_{ij}^* = \frac{x_{ij}}{\sum_i x_{ij}} \tag{5}$$

其中 x_{ij}^* 是标准化后的数据。

Step 2 计算差值假设 x_0 为参考序列。根据标准化后的数据，计算参考序列与另一序列的差值。

$$\Delta_{ij} = |x_{0j}^* - x_{ij}^*| \tag{6}$$

Step 3 计算灰色关联系数

$$\xi_{ij} = \frac{\min(\Delta_{ij}) + \rho \max(\Delta_{ij})}{\Delta_{ij} + \rho \max(\Delta_{ij})} \tag{7}$$

这里 ρ 是分辨系数，这里取 0.5。求得最终的关联度为：

$$\xi = \frac{\sum_i \xi_{ij}}{n} \tag{8}$$

图 6: 相关度较高和较低的单品样例

单品名称 1	单品名称 2	相关度
云南生菜	高瓜 (1)	0.7981
云南油麦菜	上海青	0.7482
菠菜	茼蒿	0.7179
黄白菜 (2)	紫茄子 (2)	-0.0648
红杭椒	芜湖青椒 (1)	-0.2378
红椒 (1)	茼蒿	-0.4796
螺丝椒	红椒 (1)	-0.5543

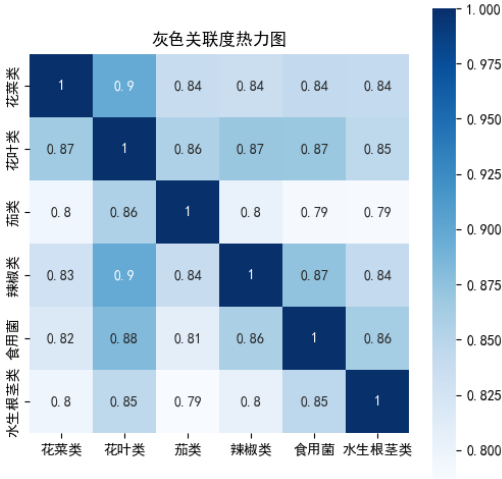


图 7: 灰色关联热力图

由图 7 可以看出茄类蔬菜与其他品类的相关性普遍较低，也与前文结果相符，这表明茄类商品往往与其他类型的商品不具有相同的趋势。

(3) 基于 FP-tree 的数据挖掘

商超中的商品往往会存在“捆绑”销售的现象。一些商品的出现往往伴随着另一些商品在同时被买下。这可能是食材的需求本存在一定的相关关系，也可能是商家的一些捆绑销售的促销策略在起作用。我们在这里认为商超一分钟之内的所有订单来自于同一顾客。为了挖掘这种相关性，我们选用 FP 树（Frequent Pattern Tree）来进行数据挖掘。FP 树算法相对于传统的 A priori 算法，避免了生成候选项集的过程，减少了计算开销，并且树状的结构使得存储数据占用空间较小。具体按照如下步骤进行数据挖掘：

Step 1 扫描数据集

对数据集进行一次扫描，统计每个项的频数，并按照频数降序排序，构建频繁项表。

Step 2 构建树

再次扫描数据集，根据频繁项表构建 FP 树。每个项的插入都遵循一个路径，如果路径上已经存在该项，则增加该节点的计数，否则创建一个新的节点。

Step 3 频繁项集挖掘

构建完成 FP 树，使用递归方法从树中挖掘频繁项集。通过从根节点开始，不断向下探索每个分支，以高效地找到频繁项集。

得到的具有强关联的组合如下：

表 9: 关联规则较强的组别

先例组合	伴随组合	组合概率
(‘红灯笼椒 (1)’, ‘芜湖青椒 (1)’, ‘螺丝椒’)	(‘红椒 (1)’,)	0.91
(‘大白菜’, ‘枝江红菜苔’, ‘牛首油菜’, ‘西峡香菇 (1)’)	(‘泡泡椒 (精品)’,)	0.91
(‘洪湖莲藕 (粉藕)’, ‘紫茄子 (2)’, ‘西兰花’, ‘金针菇 (盒)’)	(‘芜湖青椒 (1)’,)	0.90
(‘净藕 (1)’, ‘小白菜’, ‘高瓜 (1)’)	(‘西兰花’,)	0.85

上表中，组合概率是指当先例出现后，伴随组合也在同一订单出现的概率。可以看到，置信度最高的组合可能是辣椒类捆绑出售案例的体现。芜湖青椒 (1)、西峡香菇 (1) 作为热卖商品在伴随组合中多次出现。

6.2 问题 2 模型建立与求解

该问题的解决包括三个部分：

- 得到近期各蔬菜品类的销售总量与成本加成定价的关系；
- 对未来 7 天的批发价格进行预测；
- 最优化商超收益建模与求解。

6.2.1 销售总量与成本加成定价的关系

如图 8 展示了销量与时间, 定价的分布示意图：

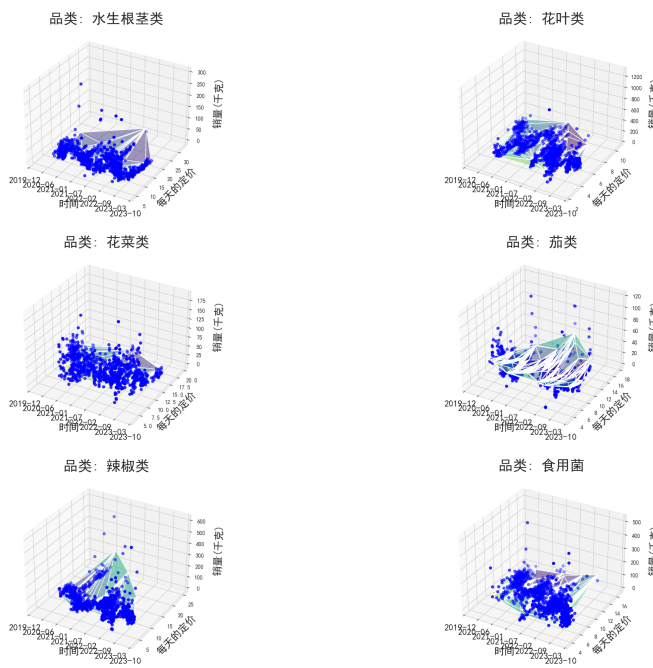


图 8: 销量与时间, 定价的关系

XGBoost(Extreme Gradient Boosting), 是一种高效的、分布式的、并行的梯度提升算法, 其核心思想是基于决策树的集成学习, 并采用梯度提升框架。与传统的梯度提升算法相比, XGBoost 在速度和性能上都有明显的优势, 还包括了正则化项, 有助于防止模型过拟合, 从而提高了模型的泛化能力^[2]。在考量销售总量与成本加成定价的关系时, 时序信息不可忽略。尤其是蔬菜订购这种时间依赖性很强的问题, 在求解近七天的定价和销售关系时, 我们想要着重捕获销量-定价关系的:

- (1) 近期的趋势;
- (2) 往年 7 月份左右的趋势;
- (3) 周期性涨落的趋势。

此外, 商超的打折销售也是决定销量的重要因素之一。在模型训练时提供每日每品类的打折占比和折扣率会有助于提升模型的性能。下面说明模型训练时的输入信息:

(1) 时序信息的独热编码和正弦余弦编码

时序信息的特征表示为 $(y, w, d, mcos, msin)$ 。我们将日期拆解为年、月、日和星期。y 特征为该天年份减去 2022; w 特征为星期数; d 特征为该天的日除以 30。对于表征月份特征, 考虑到月份在年的交界处有间断, 且以月为单位更能体现问题的季节依赖性, 因此这里将月份进行适量放缩后进行正弦余弦编码, 进而得到 mcos, msin 的特征。

(2) **打折特征**的表示折扣信息的特征表示为 (dp, dr) ，其中 dp 代表打折占比，其定义为该天该品类所有打折商品的销售量除以该天该品类的总销售量； dr 代表打折率，是该天该品类所有商品打折程度关于销售量的加权平均。

下面介绍模型的训练。训练的输入为 $(price, feature, y, w, d, mcos, msin, dp, dr)$ ，其中 $feature$ 是品类名， $price$ 是该品类当天对应的定价，训练对应的标签是该天该品类的销量。最终的误差采用平均绝对误差 (MAE)，衡量模型预测值与实际观测值之间的平均绝对差异。

我们训练的模型参数如下：

表 10: 模型参数与表现

树的深度	树的个数	学习率	MAE
6	450	0.09	15.70
6	450	0.1	15.99

训练集标签的均值为 72.75，平均相对误差为 21.6%.



图 9: 2023-7-1 花菜类预测的销量和定价关系

6.2.2 批发价格的预测模型

由于超商在进货之前不知道批发价，对于各品类批发价的预测是有必要的。本小节在对各品类的历史（加权）批发价的时序性质进行考察的前提下，确定最适配的时序预测模型，对批发价格进行预测。下文以花菜类批发价为例，考虑三年来花菜类批发价的序列数据。

下图展现了研究该时间序列的流程图：

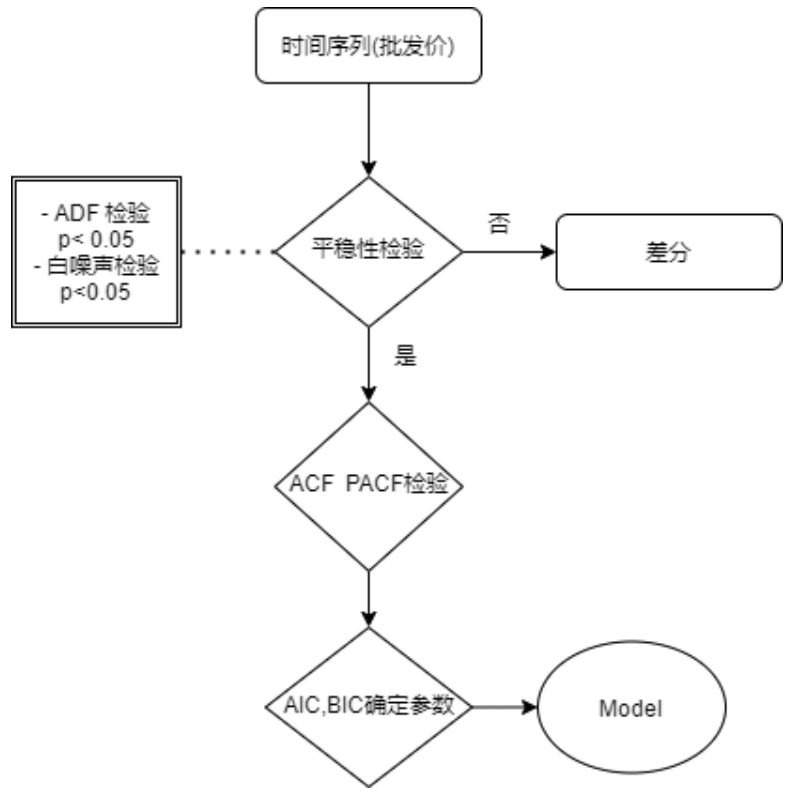


图 10: 时间序列分析流程图

(1) 平稳性检验

平稳时间序列更容易建立模型和进行准确的预测。在时间序列分析中，ADF 检验和白噪声检验通常结合使用，以确保时间序列数据满足平稳性的基本要求。ADF 检验用于检验是否存在单位根（即非平稳性），而白噪声检验用于验证模型的残差是否表现出随机性和独立性。这两种检验方法的组合有助于确保时间序列分析的可靠性。

具体到本题，白噪声检验 p 值为： $0.0000 < 0.05$ ，拒绝原假设，该序列是非白噪声序列；ADF 检验 p 值为： 0.0003 序列为平稳序列。在这种情况下，ADF 检验认为序列是平稳的，白噪声检验认为序列的残差存在自相关性。

(2) ACF, PACF 检验

下面基于 ACF 图和 PACF 图选定时序预测的模型。ACF 图显示了时间序列与其自身滞后版本之间的自相关关系，用于帮助识别时间序列中的自相关性。若 ACF 图在滞后项上迅速下降并趋于平稳，则可能需要使用自回归（AR）模型；如果是拖尾型，即自相关系数在滞后项上逐渐减小但没有截断，则可能需要使用移动平均（MA）模型；若 ACF 图同时在滞后项上呈现周期性和拖尾型，则可能需要使用季节性模型^[1]。

PACF 图的主要用途是帮助确定时间序列中的滞后项数。PACF 图在滞后项上截断为零，而非拖尾型，通常指示可以使用 AR 模型，其滞后项数由 PACF 图的截断点决定；如果 PACF 图在滞后项上存在明显的季节性截断，可能需要考虑季节性模型。花菜类批发价的 ACF, PACF 图如下：

观察 ACF 呈拖尾型，PACF 呈断尾型。依据上述论述，我们考虑自 AR 模型。在时序模型的选择中，BIC

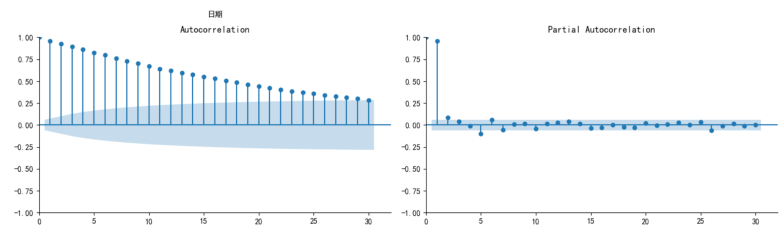


图 11: 花菜类批发价的序列数据的 ACF 图和 PACF 图

(贝叶斯信息准则) 是一种常用的模型选择准则，用于评估不同模型的相对拟合优劣。一个模型的 BIC 与模型的最大对数似然函数值、参数数量和样本数量有关。为进一步确定 AR 阶数，我们依照 BIC 准则，在 1-10 内遍历后确定取 $p=2$ 阶时 BIC 最小。基于 $p=2$ 阶的 AR 模型，我们得到 6 个品类 7 月 1-7 日的批发价的估计数据如下：

表 11: 各品类 7 月 1-7 日批发价预测

品类	7.1	7.2	7.3	7.4	7.5	7.6	7.7
花菜类	7.86	7.802	7.747	7.693	7.642	7.592	7.544
辣椒类	3.724	3.752	3.777	3.803	3.828	3.853	3.877
花叶类	3.209	3.188	3.203	3.205	3.211	3.216	3.221
茄类	4.693	4.731	4.755	4.783	4.809	4.834	4.859
食用菌	4.281	4.562	4.636	4.758	4.843	4.922	4.988
水生根茎类	12.164	12.014	11.988	11.750	11.661	11.519	11.437

在此基础上，进一步地对所得模型结果残差进行白噪声检验残差序列是否为白噪声序列，若是，残差已经无信息可提取。经编程求解残差白噪声检验得到 p 值为： $0.938 > 0.05$ ，无法拒绝原假设, 接受残差序列是白噪声序列这一假设，这意味着自相关信息提取较为充分。

6.2.3 基于遗传算法的商超收益优化模型建立与求解

(1) 确定决策变量

- α_i : 品种 i 定价
- k_i : 品种 i 补货量
- β_i : 品种 i 折扣率

(2) 确定优化目标

$$\max_{\alpha_i, k_i, \beta_i} W = \sum_{i=1}^6 [(\alpha_i(1 - \varepsilon) + \alpha_i\beta_i\varepsilon)s_i(1 + \zeta_i) - c_i k_i]$$

ε 是损耗率,

确定约束条件

我们对于待优化的变量进行了一定的范围限制, α_i^1 和 α_i^2 分别是近一周该品类定价范围加权定价最小值的 0.75 倍和最大值的 1.25 倍, k_i^1 和 k_i^2 的定义同理。这样做是考虑到 XGBoost 模型的训练集输入的范围有限, 一旦离开训练集输入范围过远, 模型可能与实际不符。

优化模型汇总

综上, 约束模型为

$$\begin{aligned} \max_{\alpha_i, k_i, \beta_i} W &= \sum_{i=1}^6 [(\alpha_i(1 - \varepsilon) + \alpha_i\beta_i\varepsilon)s_i(1 + \zeta_i) - c_i k_i] \\ s.t. &\begin{cases} \alpha_i^1 \leq \alpha_i \leq \alpha_i^2 \\ k_i^1 \leq k_i \leq k_i^2 \end{cases} \end{aligned}$$

6.2.4 模型求解

(1) 遗传算法简介

遗传算法是一种启发式优化算法, 灵感来源于自然选择的过程。这种算法模拟了自然界中生物的进化过程, 通过选择、交叉和变异等操作来产生新的解决方案^[3]。遗传算法的主要优势在于其能够在大型和复杂的搜索空间中找到优化解, 而不容易陷入局部最优。

在我们的模型中, XGBoost 模型刻画的销量与定价的关系难以刻画, 而遗传算法则可以处理此类多变量的复杂函数的优化问题, 并且在并不明确解空间的情况下, 防止陷入局部最优解。

采用遗传算法求解, 收敛情况如下图:

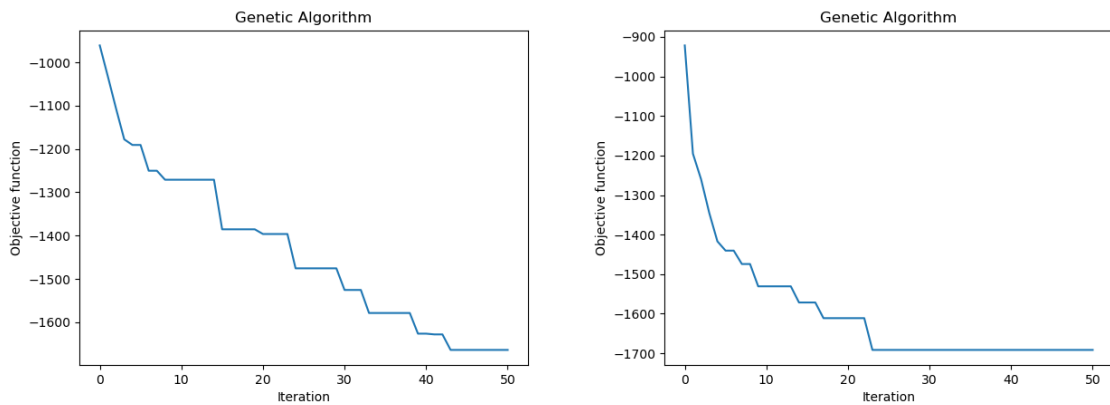


图 12: GA 收敛情况

基于上述目标函数和约束, 以 7 月 1 日为例, 优化后的参数如下:

下表展示了 7 月 1-7 号优化后的盈利与 6 月 24-30 日的盈利对比。

可见, 优化后商超日利润相比之前有大幅度提升。商家还可以有意识调整定价以及在打折的时候考虑打折的力度, 以优化商超的整体利润。

表 12: 7 月 1 日优化后参数

品类	定价	补货量
水生根茎类	22.159	23.590
花叶类	6.404	198.454
花菜类	17.273	34.461
茄类	7.106	18.110
辣椒类	7.577	111.088
食用菌	7.513	31.574

表 13: 优化前后一周盈利对比

	7.1	7.2	7.3	7.4	7.5	7.6	7.7
优化前	794.64	544.74	569.01	581.95	582.95	736.95	768.03
优化后	1745.92	1664.49	1670.72	1566.97	1584.56	1607.92	1860.87

6.2.5 基于时间衰减折扣模型的附加定价策略

在商超出售的蔬菜当天里新鲜度会逐渐下降，故除了运输等其他情况损耗的蔬菜需要打折出售以外，陈放过久的蔬菜也应该打折出售。假设品类 i 的新鲜度 K 的时间指数衰减模型如下（初始新鲜度为 1）

$$K = \exp(-\lambda t) \tag{9}$$

t 为距离早上 3 点的分钟数。这里认为对于每个品类的衰减率不一样，但是因为新鲜度衰减开始打折的阈值一样，均为默认值 0.6。下面考察近一个月各个品类最早的晚于 16:00 的打折出售时间，拟合出对应的 λ 值，再根据 $0.6 = \exp(-\lambda t)$ 得到每个品类的建议开始打折时间，特别的茄类最近一周没有打折记录，故基此无法计算建议打折时间。

表 14: 各品类的建议打折时间

品类	时间
水生根茎类	19:33
花叶类	18:47
花菜类	20:02
辣椒类	19:18
茄类	-
食用菌	18:44

在这个基础之上结合当日的优化后的折扣率可以对日内的销售情况起到一定优化作用。

6.3 问题 3 品类优化模型建立

针对销量的预测,我们仍采用 XGBoost 模型对销量进行刻画,相较于第二问数据,我们对单品数据额外增加了品类特征训练,模型 MAE=4.27,拟合较好。

针对批发价预测,49 个品种均通过了非白噪声测试和 ADF 测试证明其时间平稳性,我们对于每个品类根据 BIC 准则确定 p 参数对 2023-7-1 的批发价进行预测,总体表现较好。

接下来我们进行模型的分析与求解:

确定决策变量

统计得,2023 年 6 月 24-30 日的可售品种集合记为 $\mathbf{D}$, 统计得 $|\mathbf{D}|=49$,

- α_i : 品种 i 定价
- k_i : 品种 i 补货量
- β_i : 品种 i 折扣率
- $\delta_i = \begin{cases} 0, & \text{if 不购买该类蔬菜} \\ 1, & \text{if 购买该种蔬菜} \end{cases}$

确定优化目标

欲使商超收益最大,如问题 2 模型分析,此处加上考虑的高斯噪声 ζ .

$$\max_{\alpha_i, k_i, \beta_i, \delta_i} W = \sum_{i=1}^{49} \delta_i [(\alpha_i(1 - \varepsilon) + \alpha_i \beta_i \varepsilon) s_i (1 + \zeta_i) - c_i k_i]$$

确定约束条件

- 要求可售单品总数控制在 27-33 个, 即 $27 \leq \sum_i \delta_i \leq 33$,
- 且各单品订购量满足最小陈列量 2.5 千克的要求 $k_i \geq 2.5$;
- 在此基础上尽量满足市场对各品类蔬菜商品需求, 即不考虑噪声补货量大于预测销量, i.e. $k_i \geq s_i$;
- 类似问题 2, $\alpha_i^1 \leq \alpha_i \leq \alpha_i^2$ $k_i^1 \leq k_i \leq k_i^2$ 。

优化模型汇总

综上, 约束模型为

$$\max_{\alpha_i, k_i, \beta_i, \delta_i} W = \sum_{i=1}^{49} \delta_i [(\alpha_i(1 - \varepsilon) + \alpha_i \beta_i \varepsilon) s_i (1 + \zeta_i) - c_i k_i] \quad (10)$$

$$s.t. \begin{cases} l_i \geq 2.5 \\ 27 \leq \sum_i \delta_i \leq 33 \\ l_i \geq s_i \\ \alpha_i^1 \leq \alpha_i \leq \alpha_i^2 \\ k_i^1 \leq k_i \leq k_i^2 \end{cases}$$

问题3 模型求解

采用遗传算法, 对于约束项, 我们对超出约束的解加入一个充分大的惩罚项进处理, 并适当调大迭代次数进行求解。

下面展示模型结果。经过计算, 2023-7-1 当天应该进货如下 29 个单品: 紫茄子、圆茄子、长线茄、青茄子、云南油麦菜 (份)、云南生菜 (份)、奶白菜、木耳菜 (份)、菠菜 (份)、云南生菜、娃娃菜、菠菜、红薯尖、菜心、云南油麦菜、螺丝椒、七彩椒 (2)、青红杭椒组合装 (份)、红椒 (2)、芜湖青椒 (1)、金针菇 (盒)、蟹味菇与白玉菇双拼 (盒)、西峡花菇 (1)、野生粉藕、红莲藕带、高瓜 (2)、洪湖藕带、高瓜 (1)、净藕 (1)。下表展现了一些商品的优化后定价、补货量和打折率:

表 15: 部分商品优化后参数

单品名称	定价	补货量
螺丝椒	19.413	2.526
紫茄子 (2)	9.551	7.300
上海青	2.552	19.941
小皱皮 (份)	5.139	8.548

下图展现了目标函数的收敛情况:

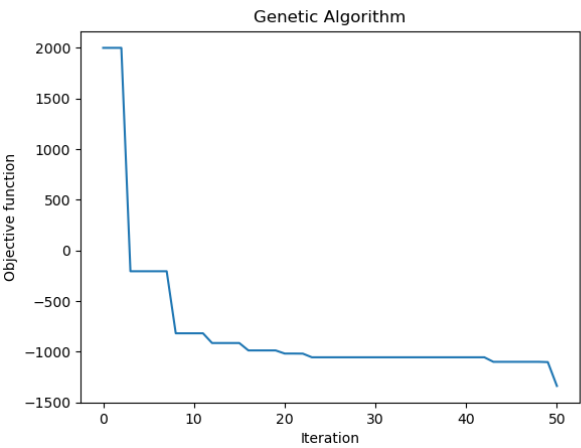


图 13: GA 收敛情况

优化后的目标函数值为 1384.537 元，相比优化前有显著提升。

6.4 问题四分析

针对问题四, 我们还可以从以下角度考虑:

- **季节/天气性数据:** 了解不同季节对销售量和价格的影响对于决策非常重要。商超可以根据近来的天气更好的预测消费者的购买需求, 针对特定季节天气的销售数据, 以更好地计划补货和定价策略, 比如部分蔬菜在温度高时销量高一些, 也有蔬菜在温度低时销量高一些。此外, 气象灾害可能对进货价格和定价有重要影响。
- **顾客流量数据:** 了解不同时间段的顾客流量可以帮助商超预测需求高峰, 从而细化补货和促销活动的安排, 如在顾客流量低的时间段进行打折等。
- **重大意外事件:** 可能需要考虑突发情况下的数据变化, 比如疫情和流行病等, 某些蔬菜价格在疫情的冲击下波动趋势十分显眼。
- **蔬菜供应链数据:** 蔬菜生产受气候等环境因素影响较大, 知道产地可以更好地了解温度等其他数据, 而如果蔬菜生产地较为单一则风险较大^[5]。
- **附近商超的销售政策流量数据:** 分析同行数据可以将其作为基准, 从而调节自己的定价和进货量以及折扣率来应对同行的竞争。
- **节假日数据:** 节假日对于商超的销售可能会有较大的影响, 比如国庆节旅游人数众多, 居家做饭的人数减少, 商超可以更多地目标用户定为餐馆酒店而不是散客购买, 以便于确定进货量和定价策略。
- **消费者数据:** 可以通过会员卡等活动获得部分消费者数据以用于客户画像, 比如性别, 年龄等数据。客户画像有利于制定更为精准的策略。如发现客户年轻人较多, 可以提前处理好蔬菜制作半成品并且收取相应的手续费, 从而满足年轻人上班较为忙碌倾向于制作简单的蔬菜的需求。

7 模型评价与改进

7.1 模型评价

7.1.1 鲁棒性分析

- 对于时间序列, 我们对其时间平稳性进行了充分检验, 并通过 BIC 准则自动调整模型参数;
- 对于机器学习模型的数据, 我们根据数据特征进行特定的处理, 包括时间数据的分离与正余弦化处理, 品种编码按照灰色关联度相近的优先顺序进行排序, 提高了模型的性能;

- 我们建模充分考虑了随机情况的影响对优化目标的影响, 即突发情况对销量的可能影响, 加入高斯噪声提高了模型的准确性;
- 打折在两个模型中都有参与, 充分考虑打折促销的影响。

7.1.2 缺点

- 即使使用了 xgboost 后, 预测的销量仍然偏高, 可见特征构建以及模型建立仍然欠妥;
- 采用遗传算法求解非线性规划可能没有得到全局最优值。

参考文献

- [1] George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and GM Ljung. *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.
- [2] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. *arXiv preprint arXiv:1603.02754*, 2016.
- [3] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T Meyarivan. Multi-objective optimization by genetic algorithms: a review. *Proceedings of IEEE International Conference on Evolutionary Computation*, 1996.
- [4] Sifeng Liu, Yingjie Yang, and Jeffrey Yi-Lin Forrest. Grey relational analysis models. *Grey Systems Analysis*, pages 77–124, 2022.
- [5] 毛莉莎. 供应链视角下蔬菜批发市场定价策略及产销模式研究. *商业经济研究*, (10):139–140, 2019.

8 附录

8.1 附录 A: 代码

```
'''
@ brief: 灰度关联分析
'''

def GRA(data, label):
    '''以 label 为母序列, 计算关联度'''
    data_normalized = data / data.mean(axis=0)
    data2 = data_normalized.drop(label, axis=1)
    y = data_normalized[label]
    # 计算关联度
    delta = np.abs(data2.sub(y, axis=0))
    b = delta.max().max()
    a = delta.min().min()
    # print(f"a:{a}, b:{b}")
    rho = 0.5 # 分辨系数, 一般取 0.5
```

```

    xi = (a + rho * b) / (delta + rho * b)

    #每列求均值
    xi_mean = cosi.mean(axis=0)

    return cosi_mean

'''

@ brief: 时间序列检测与AR算法
'''

def white_noise_detection(time_series, alpha=0.05, lags = 1):
    """
    @ :检测时间序列是否为白噪声。

    @ paras:
    - time_series: 输入的时间序列数据 (list、numpy数组或pandas Series) 。
    - alpha: 显著性水平, 通常取0.05, 表示95%的置信水平。
    """

    # 进行单位根检验 (Augmented Dickey-Fuller test)
    p_value = acorr_ljungbox(time_series, lags=1)
    p = p_value['lb_pvalue'][1]
    # 使用显著性水平来判断结果
    # print('p值为: ', p)

    if p < alpha:
        print('p值为: ', p, '小于0.05, 拒绝原假设, 该序列是非白噪声序列')
    else:
        print('p值为: ', p, '大于0.05, 无法拒绝原假设, 该序列是白噪声序列',
              )

def ADF_test(data):
    '''

    @brief : ADF检验

    @param : data:时间序列

    @ return : p值

    '''

    p_value = ADF(data)[1]

    if p_value < 0.05:

```

```

        print('p值为: ', p_value, '序列为平稳序列')
    else:
        print('p值为: ', p_value, '序列为非平稳序列')

model = AutoReg(time_series, lags = 6).fit()

forecast_days = 7
predict_y = model.predict(start=len(time_series), end=len(time_series)+
    forecast_days-1, dynamic=False)

# 有效性检验
# 1. 残差序列是否为白噪声序列, 若是, 残差已经无信息可提取。
resid = model.resid
white_noise_detection(resid)

'''
@ brief: XGBoost
'''

import numpy as np
import pandas as pd
import xgboost as xgb
from sklearn.metrics import mean_absolute_error
from xgboost import XGBRegressor
from sklearn.metrics import r2_score
from sklearn.model_selection import train_test_split
X_full = pd.read_csv('../input/training.csv')

X_full['type'] = X_full['type']/10

X_full.dropna(axis=0, subset=['sale'], inplace=True)
y = X_full['sale']
X_copy = X_full.drop(['sale', 'month_cos'], axis=1)

# Break off validation set from training data

```



```

X_train_full, X_valid_full, y_train, y_valid = train_test_split(X_copy, y
,

                                                                    train_size
                                                                    =0.8,
                                                                    test_size
                                                                    =0.2,
                                                                    random_state
                                                                    =0)

def my_model(n_estimators = 500 ,learning_rate =0.1,max_depth = 6):
    model = XGBRegressor(n_estimators=n_estimators, learning_rate=
        learning_rate, max_depth=max_depth)
    model.fit(X_train_full, y_train)
    preds = model.predict(X_valid_full)
    # 计算 $R^2$ 

    r2 = r2_score(y_valid, preds)
    print('R^2:', r2)

    mae = mean_absolute_error(y_valid, preds)
    print('MAE:', mae)
    return model

model = my_model()
'''
@ brief: 下面为调参代码
'''
list1 = []
list2 = []
list3 = []
n_estimators_list = [100,200,300,400,500,600,700,800,900]
learning_rate_list = [0.03,0.05,0.08,0.1,0.15,0.2,0.25,0.3]
max_depth_list = [4,5,6,7,8,9]
def estimator():

```

```

    for n in n_estimators_list:
        list1.append(my_model(n_estimators=n))
    sns.lineplot(x=n_estimators_list, y=list1)
    plt.title('n_estimators')
    plt.xlabel('n_estimators')
    plt.ylabel('mae')
    plt.show()

def rate():
    for n in learning_rate_list:
        list2.append(my_model(learning_rate=n))
    sns.lineplot(x=learning_rate_list, y=list2)
    plt.title('learning_rate')
    plt.xlabel('learning_rate')
    plt.ylabel('mae')
    plt.show()

def depth():
    for n in max_depth_list:
        list3.append(my_model(max_depth=n))
    sns.lineplot(x=max_depth_list, y=list3)
    plt.title('max_depth')
    plt.xlabel('max_depth')
    plt.ylabel('mae')
    plt.show()

'''
@brief: GA遗传算法
'''

import numpy as np
import pandas as pd
import xgboost as xgb
from geneticalgorithm import geneticalgorithm as ga
import os

```



```

    y = model.predict(arr)
    return y
def target(x):
    '''
    @param x: the input vector 包含所有决策变量
    alpha: 定价
    k: 采购量
    beta: 促销力度
    gain: 利润
    '''
    alpha=x[0:6]
    zeta = np.random.normal(loc=0, scale=0.05, size=6)
    k = x[6:12]
    beta=x[12:18]
    gain=0
    name_list=['水生根茎类','花叶类','花菜类','茄类','辣椒类','食用菌']
    for i in range(6):
        type_str = name_list[i]
        if type_str in varepsilon_d:
            s = get_s(alpha[i], type_str, varepsilon_d[type_str][0], beta[i])
        else:
            # 处理键不存在的情况, 例如设置默认值或引发异常
            print("key not found")
            s = 0 # 设置一个默认值, 可以根据需要进行修改
        gain += ( (alpha[i]*(1- varepsilon_d[type_str][0]))+alpha[i]*
            varepsilon_d[type_str][0]*beta[i] )*min(s,k[i])*(1+zeta[i])-c[
            i+n*6]*k[i]
    return -gain
n=6
varbound
= [[14,18], [4.78,5.35], [11.51,14], [6.36,8.1], [5.54,6.3], [4.61,6.54], [10.384,22.9

```

```

for i in range(12):
    varbound[i][0]=varbound[i][0]*0.75
    varbound[i][1]=varbound[i][1]*1.25
for i in range(6):
    varbound.append([0.75,1])
varbound=np.array(varbound)
vartype=np.array(['real']*6,['real']*6,['real']*6)).reshape(18,1)
algorithm_param = {'max_num_iteration': 50,\
                    'population_size':100,\
                    'mutation_probability':0.1,\
                    'elit_ratio': 0.01,\
                    'crossover_probability': 0.5,\
                    'parents_portion': 0.3,\
                    'crossover_type':'uniform',\
                    'max_iteration_without_improv':None}
model2 = ga( function= target , dimension=18 ,variable_type_mixed=
    vartype,variable_boundaries=varbound,algorithm_parameters=
    algorithm_param)
model2.run()

solution = model2.output_dict

```

8.2 附录 B: 结果

各品类销量多项式拟合（3 阶）

分类名称	系数 1	系数 2	系数 3	系数 4
水生根茎类	3.758e+01	-1.244e-02	6.881e-06	-8.269e-10
花叶类	2.696e+02	-1.177e-01	3.540e-05	-2.955e-09
花菜类	6.655e+01	-3.321e-02	1.034e-05	-9.591e-10
茄类	3.257e+01	-8.740e-03	1.803e-06	-1.205e-10
辣椒类	9.633e+01	-4.968e-02	1.810e-05	-1.534e-09
食用菌	9.132e+01	-2.688e-02	5.689e-06	-2.018e-10

系数依次是 0,1,2,3 次，自变量是距离 2020-7-1 的天数。

部分单品销量多项式拟合（3 阶）

分类名称	系数 1	系数 2	系数 3	系数 4	R^2 值
鲜藕带(袋)	-3.608e-02	3.451e+01	-1.101e+04	1.170e+06	0.803
快菜	3.013e-02	-3.686e+00	1.499e+02	-2.015e+03	0.775
鲜粽叶	-2.199e-03	4.606e+00	-3.216e+03	7.483e+05	0.722
丝瓜尖	6.466e-08	-1.159e-04	5.996e-02	-7.447e+00	0.684
红薯尖(份)	-2.551e-05	5.463e-02	-3.877e+01	9.128e+03	0.667

系数依次是 0,1,2,3 次，自变量是距离 2020-7-1 的天数。

图 14: ans1

FP-tree ans ₂	先例组合 [⊂]	伴随组合 [⊂]	组合概率 [⊂]
	('芜湖青椒(1)', '茼蒿', '黑油菜') [⊂]	('西峡香菇(1)',) [⊂]	0.875 [⊂]
	('茼蒿', '西峡香菇(1)', '黑油菜') [⊂]	('芜湖青椒(1)',) [⊂]	0.875 [⊂]
	('红灯笼椒(1)', '芜湖青椒(1)', '螺丝椒') [⊂]	('红椒(1)',) [⊂]	0.909090909 [⊂]
	('上海青(份)', '净藕(1)', '青尖椒(份)') [⊂]	('芜湖青椒(1)',) [⊂]	0.909090909 [⊂]
	('奶白菜(份)', '芜湖青椒(1)', '随州泡泡青') [⊂]	('金针菇(盒)',) [⊂]	0.846153846 [⊂]
	('净藕(1)', '奶白菜(份)', '菜心(份)', '金针菇(盒)') [⊂]	('芜湖青椒(1)',) [⊂]	0.916666667 [⊂]
	('净藕(1)', '菜心(份)', '西峡香菇(1)', '金针菇(盒)') [⊂]	('芜湖青椒(1)',) [⊂]	0.8125 [⊂]
	('上海青(份)', '云南油麦菜(份)', '茼蒿') [⊂]	('芜湖青椒(1)',) [⊂]	0.833333333 [⊂]
	('净藕(1)', '小白菜', '高瓜(1)') [⊂]	('西兰花',) [⊂]	0.846153846 [⊂]
	('净藕(1)', '小青菜(份)', '菠菜(份)', '西兰花') [⊂]	('芜湖青椒(1)',) [⊂]	0.857142857 [⊂]
	('小青菜(份)', '枝江青梗散花', '西兰花', '西峡香菇(1)') [⊂]	('芜湖青椒(1)',) [⊂]	0.857142857 [⊂]
	('小米椒', '红杭椒', '红椒(1)', '西峡花菇(1)') [⊂]	('芜湖青椒(1)',) [⊂]	0.833333333 [⊂]
	('洪湖莲藕(粉藕)', '紫茄子(2)', '红椒(1)', '西峡花菇(1)') [⊂]	('芜湖青椒(1)',) [⊂]	0.846153846 [⊂]
	('云南生菜(份)', '保康高山大白菜', '螺丝椒(份)', '金针菇(盒)') [⊂]	('西兰花',) [⊂]	0.857142857 [⊂]
	('云南油麦菜(份)', '奶白菜(份)', '西峡香菇(1)', '金针菇(盒)') [⊂]	('芜湖青椒(1)',) [⊂]	0.8 [⊂]
	('云南油麦菜(份)', '奶白菜(份)', '西兰花', '西峡香菇(1)') [⊂]	('芜湖青椒(1)',) [⊂]	0.8125 [⊂]
	('云南油麦菜(份)', '净藕(1)', '奶白菜(份)', '小米椒(份)') [⊂]	('芜湖青椒(1)',) [⊂]	0.909090909 [⊂]
	('云南油麦菜(份)', '奶白菜(份)', '小米椒(份)', '西兰花') [⊂]	('芜湖青椒(1)',) [⊂]	0.8125 [⊂]
	('云南油麦菜(份)', '净藕(1)', '奶白菜(份)', '螺丝椒(份)') [⊂]	('西兰花',) [⊂]	0.833333333 [⊂]
	('云南油麦菜(份)', '云南生菜(份)', '奶白菜(份)', '芜湖青椒(1)', '螺丝椒(份)') [⊂]	('西兰花',) [⊂]	0.833333333 [⊂]

图 15: ans₂

7 月 1 日优化后参数			
品类	定价	补货量	折扣率
花叶类	22.159	23.590	0.824
辣椒类	6.404	198.454	0.852
食用菌	17.273	34.461	0.843
水生根茎类	7.106	18.110	0.965
花菜类	7.577	111.088	0.960
茄类	7.513	31.574	0.811
7 月 2 日优化后参数			
品类	定价	补货量	折扣率
花叶类	19.445	24.655	0.966
辣椒类	6.511	186.808	0.849
食用菌	17.458	33.736	0.843
水生根茎类	10.014	26.237	0.781
花菜类	7.372	102.443	0.794
茄类	4.042	27.229	0.84
7 月 3 日优化后参数			
品类	定价	补货量	折扣率
花叶类	19.319	26.226	0.829
辣椒类	6.637	189.162	0.855
食用菌	15.379	28.984	0.787
水生根茎类	9.697	27.079	0.868
花菜类	7.217	94.79	0.81
茄类	8.131	30.991	0.992
+	7 月 4 日优化后参数		
	品类	定价	补货量
	花叶类	21.917	26.951
	辣椒类	6.635	172.039
	食用菌	16.213	32.174
	水生根茎类	7.429	20.547
	花菜类	6.755	108.713
	茄类	7.82	27.051

图 16: ans2₁

7月5日优化后参数			
品类	定价	补货量	折扣率
花叶类	21.216	27.642	0.81
辣椒类	6.509	191.973	0.831
食用菌	16.361	32.284	0.911
水生根茎类	9.371	19.704	0.785
花菜类	7.22	107.695	0.782
茄类	7.316	26.786	0.99
7月6日优化后参数			
品类	定价	补货量	折扣率
花叶类	21.231	27.205	0.961
辣椒类	6.563	188.259	0.774
食用菌	16.789	33.574	0.857
水生根茎类	9.589	18.393	0.754
花菜类	7.535	104.284	0.861
茄类	7.614	40.841	0.949
7月7日优化后参数			
品类	定价	补货量	折扣率
花叶类	20.526	28.178	0.942
辣椒类	6.639	191.126	0.93
食用菌	17.293	32.631	0.912
水生根茎类	9.656	27.951	0.799
花菜类	7.794	113.062	0.98
茄类	7.15	28.983	0.917

图 17: ans2₂

问题三 Ans

单品	定价	补货量	折扣率	是否购进
紫茄子(1)	2.808715258	17.0766372	0.774880886	1
圆茄子(2)	11.31733614	4.009734232	0.784445005	1
长线茄	3.928125105	10.30874642	0.777665763	1
紫茄子(2)	9.551032644	7.300155575	0.827349862	0
青茄子(1)	5.650806145	4.986032463	0.90075546	1
枝江青梗散花	3.695243029	8.353548014	0.792993125	0
西兰花	8.097710069	4.412589142	0.981757566	0
云南油菜(份)	14.9107946	2.620711326	0.996440072	1
云南生菜(份)	6.934061805	10.77188	0.961771623	1
奶白菜	5.857849075	8.098010727	0.914259388	1
小青菜(1)	9.256745762	4.001500719	0.94004127	0
木耳菜(份)	3.703820886	6.929185941	0.960654956	1
菠菜(份)	5.377136216	7.494332249	0.850827541	1
竹叶菜	3.156074684	21.02748519	0.771163054	0
上海青	2.552331251	19.94166372	0.879813899	0
茼蒿	6.295339679	14.67739843	0.796566152	0
云南生菜	3.044274982	2.85967824	0.931467141	1
外地茼蒿	9.527053157	10.16472916	0.919216718	0
娃娃菜	6.011358463	13.03086973	0.87945767	1
菠菜	14.04756335	7.740190242	0.861771789	1
红薯尖	9.478508173	2.904835979	0.944034258	1
菜心	13.02868019	3.549203375	0.937221819	1
木耳菜	11.31066261	2.659981471	0.818722519	0
云南油菜	22.15945564	2.6990676	0.7685981	1
螺丝椒	19.41321088	2.525852102	0.884937345	1
七彩椒(2)	12.90630506	4.780335178	0.773234672	1
螺丝椒(份)	6.325099245	10.43358	0.975582989	0
小皱皮(份)	5.139066962	8.54805976	0.987038653	0
青线椒(份)	9.204986451	7.449368418	0.861685367	0
青红杭椒组合装(份)	10.79573668	2.883536499	0.798181	1
姜蒜小米椒组合装(小份)	4.371085706	2.663263368	0.75367328	0
红椒(2)	11.94725112	9.427039425	0.918458613	1
芜湖青椒(1)	6.182526254	20.68955659	0.918758842	1
小米椒(份)	3.246292032	23.16371653	0.923221392	0
海鲜菇(包)	2.487860051	14.89322629	0.773675613	0
金针菇(盒)	4.656381779	2.905114954	0.896832556	1
鲜木耳(份)	4.421845072	21.06523252	0.92003212	0
虫草花(份)	21.83513386	2.596218253	0.960380186	0
白玉菇(袋)	3.415776943	8.741920652	0.999294912	0
双孢菇(盒)	4.079286306	3.777309368	0.764701165	0
蟹味菇与白玉菇双拼(盒)	19.29541469	4.191936895	0.90501813	1

图 18: ans3

西峡花菇(1)	20.0478351	7.153938834	0.896981027	1
野生粉藕	7.8447061	2.94122216	0.818854136	1
红莲藕带	2.830111093	4.734822561	0.885875911	1
高瓜(2)	2.927337823	3.071027394	0.836031603	1
洪湖藕带	2.738673267	6.25054835	0.821431863	1
菱角	4.797691275	2.868885305	0.759344585	0
高瓜(1)	1.095336964	20.86481817	0.817240039	1
净藕(1)	2.593918776	10.422193	0.894614517	1

图 19: ans3₂