

蔬菜类商品的自动定价与补货决策

摘要

蔬菜类商品具有易变质、生命周期短的特点。为了减少因品相变差成本损耗而导致的利润的损失下降，需要根据分析蔬菜价格和销量的波动趋势，为补货和定价决策做出科学参考已成为焦点问题。

问题一：首先对数据进行预处理和分类整合，绘制饼图和时间序列图。通过将数据可视化，以观察到不同品类蔬菜销售量的分布规律，其中花叶类蔬菜销售量占比最高为42%。以不同单品蔬菜（蔬菜单品）的平均销量和销售天数为指标，利用 **K-means++** 算法，可以将不同单品蔬菜分为**长期高销售额、长期低销售额、季节性高峰销售和低销售四种类型**。检验发现，六种品类蔬菜每天的销售量（随时间的变化）不满足正态性，因此采用**斯皮尔曼相关系数**计算其相关关系。

问题二：分析发现销量和定价之间不存在明显的线性关系，因此参考经济学种的价格弹性理论模型构建**反比关系**。首先根据数据的差异性和特性，选取不同的模型进行预测。例如存在明显季节因素的数据，采用 **SARIMA 模型预测**（季节性变化的），调参可得到最优的模型之一为 $SARIMA(1,1,1) \times (0,1,1)_7$ ，通过检验发现参数显著性 t 检验较好。通过准指数规律检验的数据采用**灰色预测 GM(1,1)**模型进行预测。由于蔬菜存在运损和品相变差的情况，本问考虑售是否打折销售问题。建立以商超总收益最大为目标函数，每个单品种类数量的和近似等于蔬菜品类的销售量为约束条件，建立模糊规划模型，求解出六种品类蔬菜在未来七天的日补货总量和定价策略，详见正文。

问题三：在问题二的基础上，构建新的指标：每个单品的利润率，追加考虑销售单品总量的约束和各单品最小陈列量的约束，进一步完善模糊规划模型，得到7月1日单品补货量和定价策略，基本策略为更多购进高利润率的单品。得到商超的最大收益为1024.3666元。完整结果详见正文。

问题四：通过收集和分析顾客消费习惯、供需关系、预期利润、外部影响因素和产品成本的，商超可以更好地制定蔬菜商品的补货和定价策略，实现收益最大化。收集顾客消费习惯和心理的数据可以帮助商超更好地了解市场需求，根据不同的消费群体制定促销策略，进而提高销售额。同时，供需分析也是非常关键的，可以通过分析市场的竞争情况和产品属性来调整产品的价格，以满足市场需求。最后，商超还应该密切关注外部因素的变化，及时调整策略以应对市场变化。

本文综合考虑商品销售与定价的差异性，分类模拟预测情况，进行一定程度的创新，在操作性和实用性上取得了不错的结果，且模型具有一定的推广性。

关键词：K-means 聚类 SARIMA 灰色预测 模糊规划

一、 问题重述

商家在与商超进行蔬菜交易时，并不知道销售的单品与进货价格，则要求商家通过对市场需求和历史数据的分析，制定合理的补货和定价决策，从而避免成本的损耗。除此之外，还需考虑对损坏的蔬菜进行折扣，避免因当日未能卖出导致的浪费，从而达到使商超获得最大利润的目标。

问题一：题目要求根据附件 2 中的数据分析不同品类中的分布规律和不同单品蔬菜的销售量之间是如何互相影响的。

问题二：题目要求根据附件 2 所给数据，求出不同品类蔬菜每天销量和定价之间的关联，再依据对这个关系的分析，求出商家在未来一周每天该如何补货和如何定价。

问题三：题目要求在考虑具体的销售条件约束下，依据问题二中的关系，给出能够使得商超利润最大化的 7 月 1 日的补货方案和定价方案。

问题四：题目要求收集更多的数据，进一步优化前两问建立的补货和定价的模型，使得补货和定价能更加准确，从而达到利润最大化。

二、 问题分析

2.1 问题一的分析

根据所给附件中的销售流水账，将数据按照不同品类和单品进行分类，得到同一品类或同一单品的蔬菜的销售流水账。然后进行数据清洗，剔除数据中的异常值。题目的要求分为两部分，第一部分是求蔬菜各品类、各单品间的销量的分布规律，第二部分则是蔬菜销量间的相关关系。针对该问，采用折线图、饼图等将数据可视化，先分析其大致趋势，后再利用相关系数求解出蔬菜各品类及单品间的相关关系。

2.2 问题二的分析

在综合分析了六种品类销售总量与成本加成定价的关系，通过 *SARIMA* 时间序列预测了不同品类蔬菜在 7 月 1 日至 7 月 7 日(2023 年)的销售总量之后，需要采取合理的定价策略和原料价格批发预估，平衡投入与销售的关系，以实现商超总收益利润最大化。

2.3 问题三的分析

因销售空间有限，商超进一步增加对单品总数、选取范围和陈列量的控制。7 月 1 日的单品选择需要在 2023 年 6 月 24 日至 30 日的可售品种内进行选取，同时各单品的补货量在满足最低要求的同时，尽可能的等于或低于当日的预估销售量(供应量小于或等于需求量)，减少货物滞留的损失，以实现商超收益最大化。

2.4 问题四的分析

在实际生活中，还需要考虑更多的现实因素、分析更多的影响因素，才能顺应市场供求关系，及时做出调整，以此更好地制定蔬菜商品的补货和定价策略，从而实现收益最大化。

三、 模型假设

1. 假设附件 2 中的数据皆真实可靠，因此附件 2 中未提及的数据则默认为当天未销售该单品的蔬菜。
2. 假设当日补货的蔬菜只能在当日进行出售，隔日无法进行销售。
3. 假设外部环境的相对稳定，顾客的需求量和蔬菜的供给量都不会发生大幅度的改变。
4. 假设商超在采用“成本加成定价”方法确立蔬菜定价时，在原料、运输、储存、

- 人工成本的基础上，进行一定利润的加成。
- 假设蔬菜的损耗情况，包括从交易到销售整个过程，除此之外不考虑其他因素造成的损失。
 - 假设各品类各单品蔬菜的销售数据（2020 年 7 月 1 日至 2023 年 6 月 30 日）可以反应商品价格的整体长期趋势。
 - 假设商超在当日确定销售价格后，不会受其他因素的影响，再进行价格的变动。

四、 符号说明

符号	说明	单位
i	单品种类数量	个
j	蔬菜品类数	个
N	天数	天
a_{ij}	j 种品类下第 i 种单品的销售价格(未受损耗部分)	元
t_i	每个单品每天的销售量	千克
e_i	每个单品平均每天销售量	千克
p_{ij}	j 种品类下第 i 种单品的损耗率	\
x_{ij}	j 种品类下第 i 种单品的进货量	千克
b_{ij}	j 种品类下第 i 种单品的销售价格(受损耗部分)	元
c_{ij}	j 种品类下第 i 种单品的批发价格	元
f_j	j 品类收益和	元
z	以品类为单位商超总收益和	元
r_{ij}	j 品类第 i 种单品的利润率	\
m_j	j 单品的收益和	元
n	以单品为单位商超总收益和	元
d_j	松弛变量	

五、 模型的建立与求解

5.1 问题一模型的建立与求解

5.1.1 数据预处理

时间序列数据的预处理对于数据分析和建模非常关键，它可以帮助我们准确地理解数据、发现异常值并为进一步的分析做好准备。

针对附件 2 中可能存在的不完整、有异常的数据，进行数据预处理。

1.数据收集与加载： 首先，收集时间序列数据并将其加载到分析环境中。确保数据的时间格式正确，并且数据字段清晰。

2.数据探索： 对数据进行初步的探索性分析，包括查看数据的统计摘要、绘制时间序列图以及观察趋势和季节性。

3.异常值检测： 使用 3-Sigma 法则（也称为 3σ 法则）来检测异常值。这个方法基于数据的均值和标准差，通常认为在均值加减 3 倍标准差范围之外的数据点是异常值。

可以使用以下步骤来检测异常值：计算时间序列数据的均值和标准差。对每个数据点，计算其与均值的偏差（ $z\text{-score}$ ），即 $(\text{数据点} - \text{均值}) / \text{标准差}$ 。根据 3-Sigma 法则，如果 $z\text{-score}$ 大于 3 或小于 -3，将该数据点标记为异常值。**异常值处理：**对于检测到的异常值，可以选择采取以下方法之一来处理它们：将异常值替换为均值、中位数或其他合适的值。

其次将蔬菜的销量按照不同品类、不同单品进行分类整合，以方便问题后续的求解。

5.1.2 模型的求解

5.1.2.1 不同品类蔬菜销售量的分类规律

首先将六种品类的蔬菜进行整合，分别求出在 2020 年 7 月 1 日至 2023 年 6 月 30 日内这六种品类蔬菜的总销售量，再将总销售量进行可视化，即绘制饼图清晰直观地表示这六种品类销量之间的分布规律。

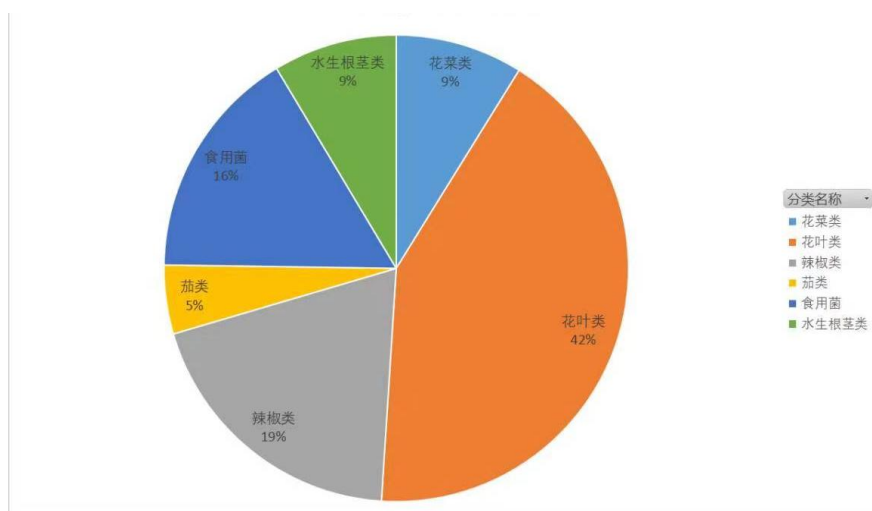


图 5.1-1 六种品类蔬菜销售量占比图

由上图可知，销量最大的是花叶类蔬菜，占总销量的 42%。其次是辣椒类蔬菜、食用菌类蔬菜，其占比分别为 19% 和 16%。水生根茎类与花菜类蔬菜的销量相近，占比均为 9%。而茄类蔬菜销量最少，占比仅为 5%。

由题所知，蔬菜供应品种在 4 月至 10 月较为丰富，由此可看出蔬菜的销售市场还与时间相关。因此，以月份为单位，做出各个品类蔬菜在每个月的销量图。

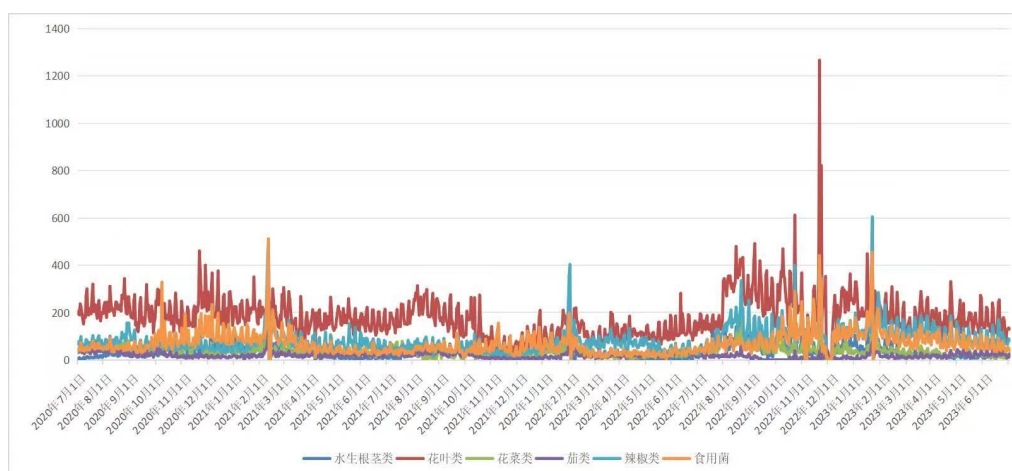


图 5.1-2 六种品类蔬菜销售量变化曲线图

将六种品类蔬菜销售量变化进行可视化，得到曲线图如上。分析上图发现蔬菜在 7 月至 10 月的销售量较高，2 月时销售量会短暂但大幅度的上升。总体销售量与季节性相关。

花菜类销售量变化曲线详见下图，其余品类蔬菜销售量变化曲线图见附录。



图 5.1-3 花菜类销售量变化曲线

5.1.2.2 不同单品蔬菜销售量的分类规律

$K\text{-means}++$ 是一种聚类算法，可以将数据点分成具有相似特征的簇。它是 $K\text{-means}$ 聚类的改进版本，旨在改善初始簇中心点的选择，以提高算法的收敛速度和聚类质量。

在 $K\text{-means}++$ 中，初始簇中心点的选择是该算法的关键创新。传统的 $K\text{-means}$ 算法通过随机选择数据点作为初始中心点，但这种方法可能导致算法陷入局部最小值或收敛速度缓慢。 $K\text{-means}++$ 通过以下学术方法来改进初始簇中心点的选择：

1. 选择第一个中心点：首先，从输入数据点中随机选择一个数据点作为第一个簇中心点。

2. 选择后续中心点：对于后续的簇中心点的选择， $K\text{-means}++$ 采用了一个概率分布的策略。具体来说，它计算了每个数据点与已选簇中心点的最短距离（即到最近簇中心点的距离），并使用这些距离的平方作为权重来选择下一个簇中心点。这个过程确保了选择离已有簇中心点较远的数据点作为新的中心点，从而更好地覆盖数据的全局结构。

通过改进初始中心点的选择， $K\text{-means}++$ 能够更快地达到收敛，减少了陷入局部最小值的风险，并提高了最终聚类的质量。它在许多实际应用中表现出良好的性能，成为 $K\text{-means}$ 聚类的首选改进版本之一。

本问中构建了两个单品的指标：单品在总共有多少天在销售，即销售天数 N 和不同

单品蔬菜的每天平均销量：
$$e_i = \frac{\sum t_i}{N}.$$

利用该算法将不同单品的销售量分为销售天数多且销售量大（长期高销售额）、销售天数多但销售量较小（长期低销售额）、销售天数少但销售量大（季节性高峰销售）、销售天数少且销售量小（低销售）四类。

表5.1-1 最终聚类中心

	1	2	3	4
销售天数	827	463	236	32
每日销售量	12.2036	11.6062	7.4524	3.3411

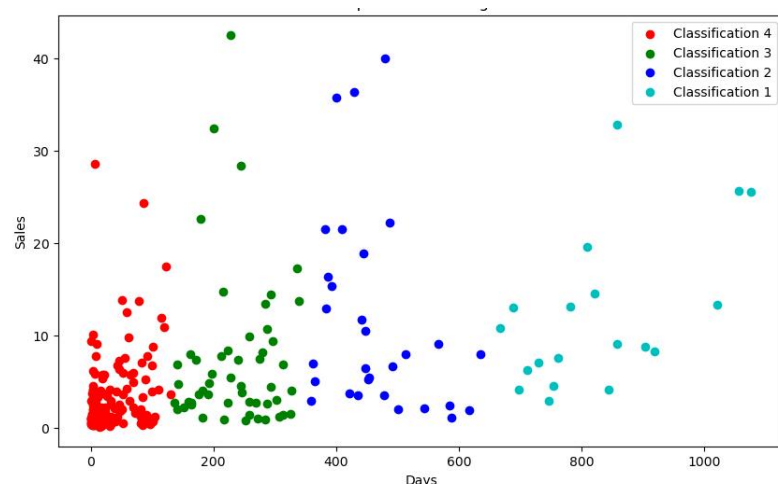


图 5.1-4 K-means 聚类散点图

西兰花、云南生菜、螺丝椒等属于销售天数多且销售量大的第一类。小白菜、杏鲍菇、圆茄子等属于销售天数多但销售量较小的第二类。外地茼蒿、萝卜叶、虫草花等属于销售天数少但销售量大的第三类。牛首生菜、冰草、菱角等属于销售天数少且销售量小的第四类中。部分数据整理后得到下表。

表 5.1-2 部分单品聚类结果表

单品编码	单品名称	销售天数	每日销售量	聚类结果
102900005115168	牛首生菜	102	8.821931373	4
102900005115199	四川红香椿	151	2.206774834	3
102900005115250	西峡花菇(1)	297	9.463690236	3
102900005115625	本地小毛白菜	47	2.574893617	4
102900005115748	白菜苔	92	7.811695652	4
102900005115762	苋菜	635	8.031592126	2
102900005115779	云南生菜	810	19.64254444	1
102900005115786	竹叶菜	667	10.85571814	1
...
102900005115793	小白菜	478	3.571763598	2

以下图部分单品销售量变化为例：

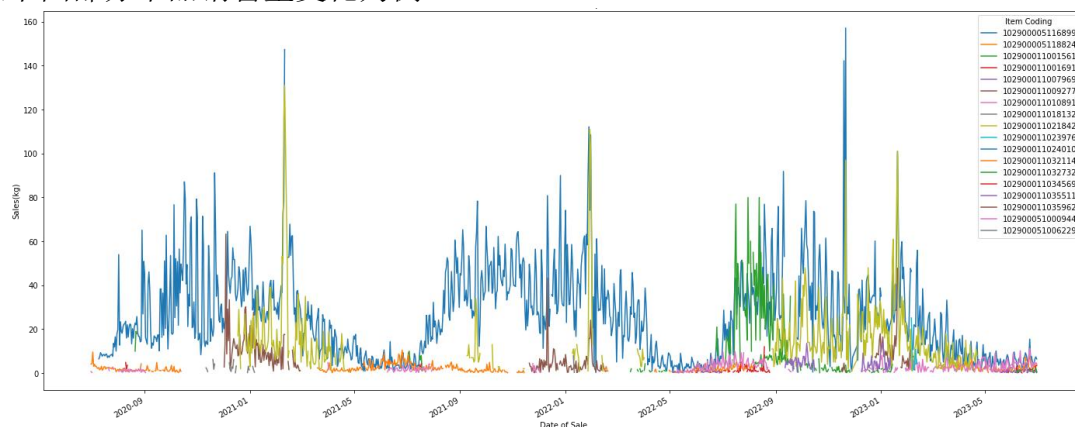


图 5.1-5 部分单品蔬菜销售量变化曲线

由上图可得到部分单品销售量变化趋势，发现多数单品蔬菜销售量呈季节性变化。

如净藕（1）（蓝色曲线），属于一类聚类即销售天数多且销售量大（长期高销售额）。高瓜（1）（橙色曲线）属于二类聚类，即销售天数多但销售量较低（长期低销售额）。莲蓬（个）（绿色曲线）属于四类聚类，即销售天数少且销售量低（低销售）。荸荠（棕色曲线）属于三类聚类（季节性销售），即销售天数较少但销售量高。

全部聚类结果绘制成散点图，完整见附录 1。

5.1.2.3 不同品类蔬菜销售量的相关关系

将附件 2 数据根据蔬菜品类进行整合，得到六种蔬菜品类每天的销售量。对其进行正态性检验，发现六种品类的蔬菜均不满足正态性，故采用斯皮尔曼相关系数计算其相关关系。详细结果如下：

表 5.1-3 六种品类蔬菜销售量相关系数表

	水生根茎类	花叶类	花菜类	茄类	辣椒类	食用菌
水生根茎类	1.000	0.454	0.412	-0.178	0.352	0.616
花叶类	0.454	1.000	0.643	0.271	0.606	0.607
花菜类	0.412	0.643	1.000	0.214	0.445	0.477
茄类	-0.178	0.271	0.214	1.000	0.126	-0.085
辣椒类	0.352	0.606	0.445	0.126	1.000	0.548
食用菌	0.616	0.607	0.477	-0.085	0.548	1.000

由上表可知，除茄类与水生根茎类、食用菌与茄类蔬菜销量间为负相关关系，其余不同品类蔬菜销量间均为正相关关系。其中食用菌与水生根茎类、食用菌与花叶类、辣椒类与花叶类之间销量的相互关系均较强，相关系数均大于 0.6。



图 5.1-6 六种品类蔬菜斯皮尔曼相关系数热力图

5.2 问题二模型的建立与求解

5.2.1 销量与定价的关系

我们首先通过绘制散点图来探究这两个变量之间的关联性。然而，散点图的结果表明销售量与定价似乎呈现出一种不明显的相关性，数据点似乎呈现出聚集的趋势。其中

一个单品的销量和定价关系散点图绘制如下：

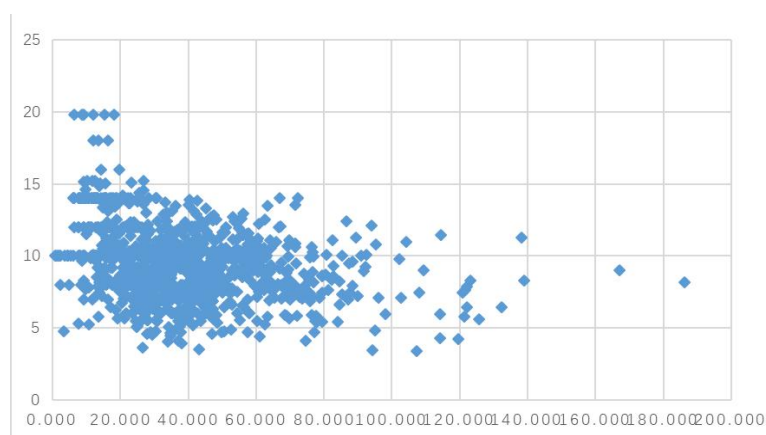


图 5.2-1 花菜类销量定价散点图

然而，对于销售量和定价这两个变量而言，由于二者都是时间序列数据，这引发了对时间维度的关联性的进一步研究的需求。考虑到时间因素可能在这一关系中扮演着关键角色，我们计划深入探讨销售数据与时间之间的相关性。时间序列分析将允许我们更全面地理解销售量和定价之间的动态性，包括趋势、季节性变化以及其他与时间相关的因素。将两个变量绘制成时序图可以得到下图：

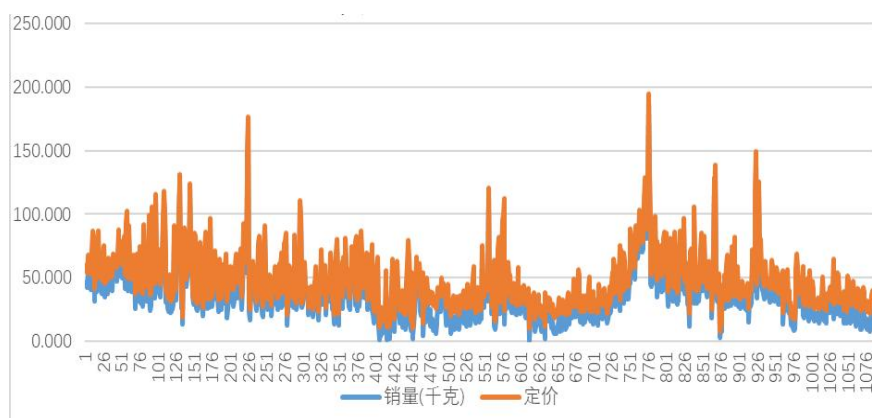


图 5.2-2 花菜类销量和定价的时序图

由上述散点图、时序图和分析可知，不同品类蔬菜销量与定价之间的拟合效果较差，则表明不存在明显的线性关系，所以参考经济学中价格弹性角度分析蔬菜销售量与定价之间的关联。

蔬菜通常被认为是价格弹性较高的商品之一。价格弹性是指商品价格变动对需求量变化的敏感程度。蔬菜的价格弹性较高意味着当价格上涨时，需求下降较多，而当价格下降时，需求增加较多。

这是因为蔬菜通常属于日常消费品，价格波动会直接影响人们的购买决策，而且蔬菜通常没有强烈的品牌忠诚度，消费者更容易根据价格做出购买选择。总体而言，蔬菜通常被认为是相对价格敏感的商品，且蔬菜的销售量和定价之间大致呈反比关系。

则各蔬菜品类的销售总量 Y 与成本加成定价 X 的关系，可以表示为：

$$Y_i * X_i = a_i$$

其中每个蔬菜品类的乘积系数结果整理得到下表

表 5.2-1 不同蔬菜品类销售量与定价的乘积系数表

蔬菜品类	a_i
花叶类	671.889
花菜类	59.765
辣椒类	414.6
茄类	135.715
食用菌类	250.508
水生根茎类	66.96

5.2.2 补货总量和定价策略制定中的时间序列预测思路

本研究旨在针对蔬菜品类在未来一周（2023 年 7 月 1 日至 7 日）的日补货总量和定价策略进行分析和预测。为了实现这一目标，首先，蔬菜品类的销售量、单品的进货价、单品的不打折销售单价以及单品的打折销售单价等因素，都是时间序列数据，其特点各异，因此需要采用不同的预测模型来处理这些数据。

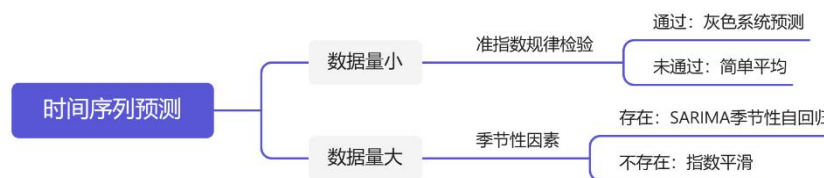


图 5.2-3 数据预测思路图

1. 销售量的预测：蔬菜销售量通常受到季节性、周几、节假日等因素的影响。为了预测未来一周的销售量，我们将采用 *SARIMA*（季节性自回归移动平均模型），*SARIMA* 模型结合了自回归（*AR*）、差分（*I*）和移动平均（*MA*）成分，以及季节性（*S*）成分，因此能够处理具有周期性模式的数据。通过选择适当的参数，对销售量数据进行建模，并生成未来一周内销售量的预测值。以考虑并捕捉这些影响因素，提供准确的销售量预测。

2. 单品的进货价预测：单品的进货价通常受供应商定价策略和市场波动等因素影响。对于数据量较多的情况，可以采用时间序列模型进行预测。

3. 单品的不打折销售单价预测：不打折销售单价可能受市场需求因素影响。对于数据量较多的情况，可以继续使用时间序列模型进行预测。但如果数据量较少，使用灰色预测（*GM*（1,1）预测）来估计未来一周内单品的不打折销售单价。

4. 单品的打折销售单价预测：打折销售单价可能受促销活动、季节性折扣等因素影响。同样，如果数据量较多，可以采用时间序列模型进行预测，但如果数据有限，使用灰色预测（*GM*（1,1）模型），以预测未来一周内单品的打折销售单价。

5.2.3 *SARIMA* 模型预测未来不同品类蔬菜的销售量

由问题一中对不同品类蔬菜销售量变化的曲线图和分布规律可知，蔬菜的供应与销售较大程度地受季节影响，且销售量会随着季节变化而发生改变，即在时间序列上表现出明显的季节性和周期性特征^[2]。故本问采用季节性回归移动平均模型（*SARIMA* 模型）对未来不同品类蔬菜的销售量进行预测。

SARIMA 模型原理

SARIMA 时间序列模型是基于常用的时间序列预测，多加考虑季节性因素。*SARIMA* 模型可表示为 *SARIMA*(p, d, q)(P, D, Q)。其中 (p, d, q) 是非季节部分，(P, D, Q)_s 是季节

部分。 p 是自回归项数, q 是移动平均项数, d 是差分次数, P 是季节自回归阶数, Q 是季节移动平均阶数, D 是季节差分次数。

假设 $\{x_t, t = 0, \pm 1, \dots\}$ 是周期为 s 的随机序列, 则 *SARIMA* 的一般形式为:

$$\Phi_p(L)A_p(L^s)(\Delta^d \Delta_s^D x_t) = \Theta_q(L)B_Q(L^s)u_t$$

其中:

$$\begin{aligned}\Phi_p(L) &= 1 - \phi_1 L - \phi_2 L^2 - \dots - \phi_p L^p \\ A_p(L) &= 1 - \alpha_1 L^s - \alpha_2 L^{2s} - \dots - \alpha_p L^{Ps} \\ \Theta_q(L) &= 1 - \theta_1 L - \theta_2 L^2 - \dots - \theta_q L^q \\ B_Q(L^s) &= 1 - \beta_1 L^s - \beta_2 L^{2s} - \dots - \beta_Q L^{Qs}\end{aligned}$$

其中 u_t 为白噪声序列。

SARIMA 模型构建

首先对不同品类蔬菜销售量的数据序列进行季节性、趋势性检验, 问题一中变化趋势图中可看出以季节为周期的变动规律。

其次对序列进行平稳性检验, 若为非平稳序列, 则可通过差分处理等转化为平稳序列。接着进行白噪声检验, 以判断该序列为非随机数据。

观察序列的自相关函数和偏向关函数, 以此为依据对模型参数进行选择与估计, 构成不同的模型结构形式。

对不同模型结构形式的预测数据进行对比与显著性检验, 从而进一步对模型进行优化。得到性能较好的模型来进行预测。

SARIMA 模型求解

由六种品类蔬菜销售量变化曲线图中看出, 蔬菜销售量呈现较为明显的季节性周期, 每年 7 月至 11 月的蔬菜销售较较多, 每年 2 月时会有明显的升高。数据在周期内大致呈现递增趋势, 即为非平稳序列。

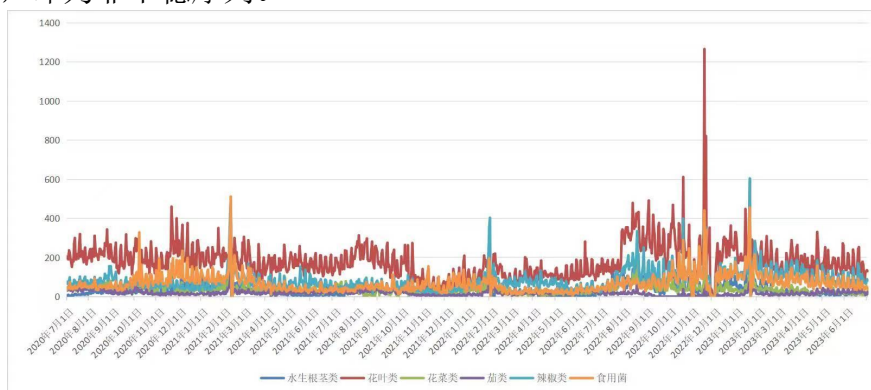


图 5.2-4 六种品类蔬菜销售量变化曲线图

以辣椒类蔬菜为例:

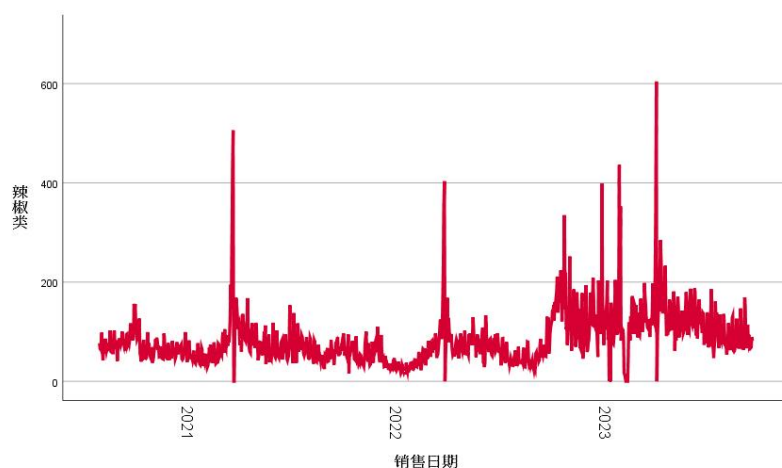


图5.2-5 辣椒类蔬菜销售量

由辣椒类蔬菜销售量变化曲线图中看出，辣椒类销售量呈现较为明显的季节性周期，呈 V 字型增长。数据在周期内大致呈现递增趋势，即为非平稳序列。

现在考虑对序列建立 $SARIMA$ 模型，做该序列的自相关图和偏自相关图，如下。

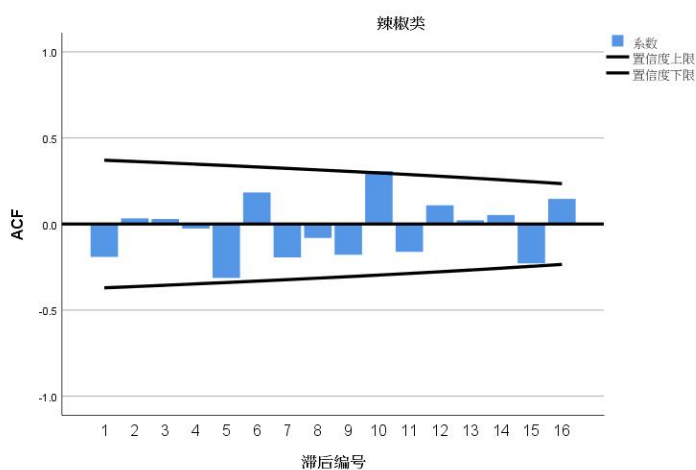


图5.2-6辣椒类蔬菜销售量自相关图

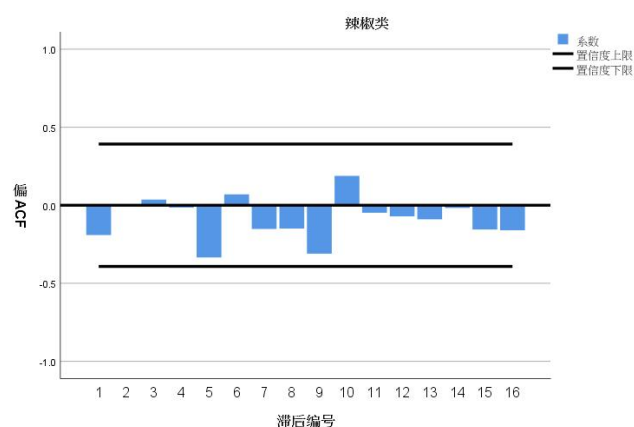


图5.2-7 辣椒类蔬菜销售量偏自相关图

从自相关图（ACF）、偏自相关图（偏 ACF）看出，大部分编号位于置信区间内部，说明序列是基本平稳的。

本问对多种模型进行拟合，得到最佳参数模型如下：

表 5.2-2 不同品类蔬菜最佳参数表

蔬菜品类	最佳参数
花菜类	$SARIMA(1,1,1) \times (0,1,1)_7$
花叶类	$SARIMA(1,0,1) \times (0,1,1)_7$
茄类	$SARIMA(1,1,1) \times (0,1,1)_7$
辣椒类	$SARIMA(1,1,1) \times (0,1,1)_7$
食用菌类	$SARIMA(1,1,1) \times (1,1,1)_7$
水生根茎类	$SARIMA(1,1,1) \times (0,1,1)_7$

发现 $SARIMA(1,1,1) \times (0,1,1)_7$ 参数显著性 t 检验结果较好，误差项的方差估计值为 0.100，对模型的残差序列检验后发现其为平稳的白噪声序列。

在该模型的预测下，不同品类蔬菜在未来七天的销售量预测值如下：

表 5.2-3 不同品类蔬菜未来七天销售量预测值表

	7月1日	7月2日	7月3日	7月4日	7月5日	7月6日	7月7日
辣椒类	201.12	185.11	135.77	130.26	132.75	133.31	155.49
食用菌类	30.58	28.90	15.09	13.89	13.36	14.01	18.41
花菜类	112.8	106.77	71.57	70.22	69.15	75.48	93.72
花叶类	28.23	28.05	21.41	19.23	19.35	19.50	23.15
茄类	67.68	65.30	43.47	40.79	46.58	44.94	52.91
水生根茎类	31.81	29.0	14.37	13.88	13.30	12.28	19.09

5.2.4 模型的建立

蔬菜批发价格的预估，通过比较销售频率和销量，可以将产品进行分类预测，以便更好的模拟预估。在预测之前，以 2020 年至 2023 年中一百种蔬菜的销量分析比较为例，发现 97% 的蔬菜销售频率超过 30 次，且交易数目比较频繁，使用 $ARIMA$ 时间序列预测的效果比较好。对于销售频率低于 30 次的商品，由于含有信息量比较少，使用灰色预测模型预测该部分蔬菜的批发价。

以下灰色预测模型的分析以花叶类的丝瓜尖为例。

步骤一：数据的检验与处理

为了保证 $GM(1,1)$ 建模方法的可行性，对丝瓜尖的历史批发价数据做必要性的检验处理，包括对数据量和负数的检测。

设该非负数据列为：

$$x^{(0)} = (x^{(0)}(1), x^{(0)}(2), \dots, x^{(0)}(n)),$$

准指数规律检验，计算数列的级比和原始序列的光滑比：

$$p(k) = \frac{x^{(0)}(k)}{x^{(1)}(k-1)}$$

经过 MATLAB 程序计算得到以下两个指标：

指标 1：光滑比小于 0.5 的数据占比为 94.4444%

指标 2：除去前两个时期外，光滑比小于 0.5 的数据占比为 100%

根据参考指标，指标 1 一般要大于 60%，指标 2 要大于 90%，可以通过检验。

若不满足条件，需要对数据进行平移变换。满足约束条件即可以建立 $GM(1,1)$ 模型进行灰色预测。

步骤二：建立 $GM(1,1)$ 模型

$$x^{(0)}(k) + a * z^{(1)}(k) = b$$

采用回归分析求得 a, b 的估计值，得到预测值：

$$\hat{x}^{(0)}(k+1) = \hat{x}^{(1)}(k+1) - \hat{x}^{(1)}(k), k=1, 2, \dots, n-1,$$

实验组与各 GM(1,1)组预测结果如下表所示：

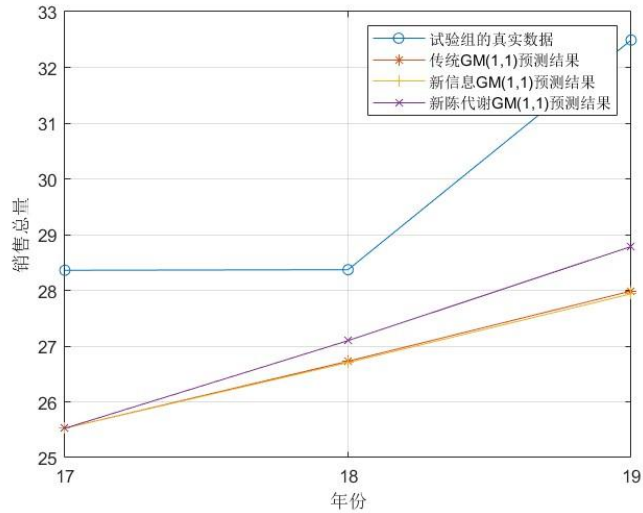


图 5.2-8 实验组与各 GM(1,1)组预测结果图

传统 GM(1,1)对于试验组预测的误差平方和为 31.0021，新信息 GM(1,1)对于试验组预测的误差平方和为 31.5251，新陈代谢 GM(1,1)对于试验组预测的误差平方和为 23.3638。新陈代谢 GM(1,1)模型的误差平方和最小，选其进行后续预测。

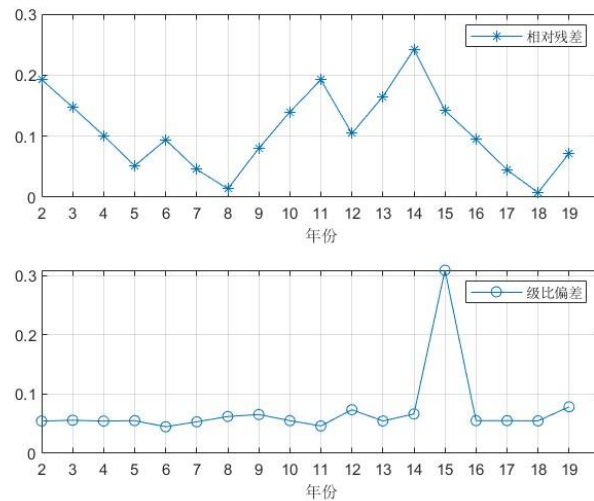


图 5.2-9 相对残差和级比偏差图

往后预测 1 期的结果为 31.8262，平均相对残差和平均级比偏差分别为 0.10725 和 0.071975，该模型对原数据的拟合效果非常好。

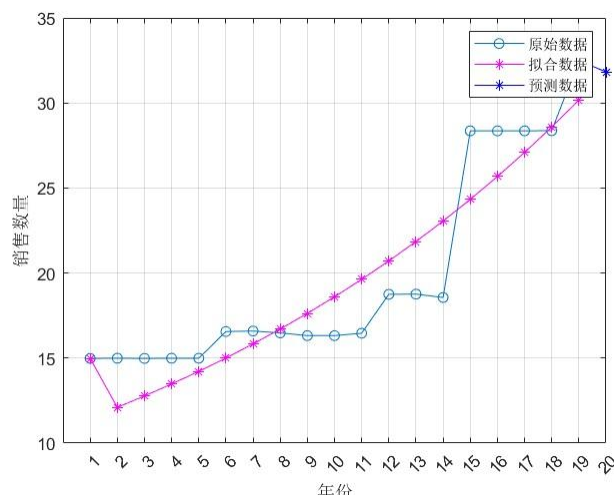


图 5.2-10 拟合效果图

在蔬菜不受损耗的前提下，为保证收益，商超在确保成本不会亏损的同时，对成本进行一定程度的加成，得到最终的销售价格。而加成系数的确定，可以通过分析商超对以往不受损耗蔬菜的销售价格和批发价格关系得到。

规划模型的建立

由于蔬菜类商品的保鲜期比较短，质量易受时间影响，商超隔日无法将剩余产品卖出。因此对于受到损耗的蔬菜，为了减少产品积压库存造成的经济损失，进行合理范围内的打折促销。

不同品类不同单品的批发价价格使用 *ARIMA* 时间序列预测，由于蔬菜商品价格变化范围只有 2020 年 7 月到 2023 年 6 月，在时间长度上不足以显示周期性，因此排除循环变动(C)的因素影响。蔬菜单品的批发价的变动具有长期趋势(T)、季节趋势(S),同时由于随机因素影响增加了不规则变动(I)，因此，本文选取的时间序列数据只考虑 I, T, S，选用乘法模型：

$$Y = T * S * I$$

目标函数：使得商超总收益最大：

$$\max w_j = \sum_{i=1}^n \left[(1 - p_{ij}) * a_{ij} * x_{ij} + p_{ij} * x_{ij} * b_{ij} - x_{ij} * c_{ij} \right]$$

商超总收益计算：j 种品类下第 i 种单品的销售额减去该单品成本价，即为该单品的收益额，j 品类下所有单品的收益和即为该品类的收益，所有品类的收益即为商超的总利润。

经过对模型的相关计算，可以将单类别收益的计算表示为该类别所有单品的利润率与进货量的乘积的求和：

$$\max f_j = r_{ij} * x_{ij}$$

商超的总收益和表示为：

$$\max z = \sum_{j=1}^6 r_{ij} * x_{ij}$$

约束条件一：每个单品种类数量的求和要近似等于或者略小于蔬菜品类的销售量：

$$\sum_{i=1}^n x_{ij} \leq [N_j, d_j]$$

因此，建立如下模糊规划模型：

$$\begin{aligned} \max z &= \sum_{j=1}^6 r_{ij} * x_{ij} \\ s.t. &\sum_{i=1}^n x_{ij} \leq [N_j, d_j] \end{aligned}$$

5.2.5 模型求解与结果分析

使用 Matlab 编程计算得到结果，经整理后得到不同品类蔬菜未来七天每天的补货量和定价，结果分别如下：

表 5.2-4 不同品类蔬菜未来七天的每天总补货量

蔬菜品类	7月1日	7月2日	7月3日	7月4日	7月5日	7月6日	7月7日
花叶类	201.12	185.11	135.77	130.26	132.75	133.31	155.49
花菜类	30.58	28.90	15.09	13.89	13.36	14.01	18.41
辣椒类	112.88	106.77	71.57	70.22	69.15	75.48	93.72
茄类	28.23	28.05	21.41	19.23	19.35	19.50	23.15
食用菌类	67.68	65.30	43.47	40.79	46.58	44.94	52.91
水生根茎类	31.81	29.05	14.37	13.88	13.30	12.28	19.08

表 5.2-5 不同品类蔬菜未来七天的每天定价

蔬菜品类	7月1日	7月2日	7月3日	7月4日	7月5日	7月6日	7月7日
花叶类	3.341	3.630	4.949	5.158	5.061	5.040	4.321
花菜类	1.954	2.068	3.960	4.303	4.474	4.267	3.247
辣椒类	3.673	3.883	5.793	5.904	5.996	5.493	4.424
茄类	4.808	4.838	6.340	7.058	7.015	6.960	5.864
食用菌类	3.701	3.837	5.763	6.141	5.378	5.575	4.735
水生根茎类	2.105	2.305	4.661	4.823	5.033	5.455	3.509

5.3 问题三模型的建立与求解

5.3.1 模型的建立

由过往的数据可知，当日制定的产品价格稳定，商超不会在当日销售量不在预估范围的情况下再进行价格的变动。

目标单品选取：

6月24日至30日的可售品种为50余种，为使得收益最大化，同时满足对单品总数的要求，将所有可售品种的利润率计算后进行排序比较，选取前27至33位产品，即为目标产品。第二问获得的单日销售价格预测模型可估计补货量。

$$\max m_j = r_{ij} * x_{ij}$$

目标函数：使得商超总收益最大。所有单品的收益和即为该商超的总收益。

$$\max n = \sum_{j=1}^h m_j, 27 \leq h \leq 33$$

约束条件 1：第二问获得的单日销售价格预测模型可估计补货量，每个单品数量的要近似等于或者略小于该单品的预估销售量。

$$x_j \leq [N_j, d_j]$$

约束条件 2：单品最低陈列量不低于 2.5kg。

$$x_j \geq 2.5$$

因此，建立如下模糊规划模型：

$$\begin{aligned} \max n &= \sum_{j=1}^{33} r_j * x_j \\ s.t. &= \begin{cases} x_j \leq [N_j, d_j] \\ x_j \geq 2.5 \end{cases} \end{aligned}$$

5.3.2 模型的求解与结果分析

使用 Matlab 编程计算得到结果，经整理后得到表格。部分展示如下，完整的日补货总量和定价策略见附录。

表 5.3-1 部分单品蔬菜的补货量和定价策略表

编号	蔬菜名称	品类	补货量	不打折销售单价	打折销售单价
10290001103234 3	七彩椒(2)	辣椒类	77.88	25.32	16.49
10290001103273 2	高瓜(2)	水生根茎类	4.96	22.71	14.80
10290005100094 4	洪湖藕带	水生根茎类	2.50	29.66	17.38
10290000511525 0	西峡花菇(1)	食用菌	38.15	21.48	19.51
10290001103507 8	红椒(2)	辣椒类	2.50	17.53	11.64
10290001100694 8	外地茼蒿	花叶类	114.77	16.16	14.20
10290001103398 2	紫茄子(1)	茄类	16.44	11.04	6.83
...
10290001103009 7	云南油麦菜(份)	花叶类	2.50	4.39	2.86

该策略主要是构建单品利润率指标

$$r_j = (1 - p_{ij}) * a_{ij} * x_{ij} + p_{ij} * x_{ij} * b_{ij} - x_{ij} * c_{ij}$$

提高对高利润率单品的进货量来提高总收益。通过上述补货和定价策略，可以得到 7 月 1 日商超的最大收益，为 1024.3666 元。

5.4 问题四的求解

在实际生活中，还需要考虑更多的现实因素、分析更多的影响因素，才能顺应市场供求关系，及时做出调整，以此更好地制定蔬菜商品的补货和定价策略，从而实现收益最大化。

- 1.顾客消费：收集不同消费群体的消费习惯、消费心理和购买能力，更好地明确产品的市场定位和目标客群，便于卖家制定促销的相关策略，使产品的销售更具有针对性。
- 2.供需分析：市场需求对产品的价格往往有很明显的影响，卖家可以通过分析产品的竞争品、销量、生命周期、竞争力、优势等属性，更好的捕获市场的变化，及时调整

产品的价格。

3.预期利润：收集更多不同商品的售价、销售额和利润的历史数据，帮助卖家在产品出售前对产品的销量和利润有一定程度的估计，可以作为补货和定价策略的参考。

4.外部因素影响：分析环境变换和政策调整，研究外界因素对于蔬菜质量和存储的影响，卖家可以及时根据变化调整补货的数量和产品的价格，减少变化引发的食用质量下降、滞销产品、积压库存、存储成本增加等问题。

5.产品成本：从生产、运输到销售，需要考虑原材料、人工、存储等方面的成本。出售高端、品质优秀的蔬菜产品，相较普通产品的蔬菜，需要付出更多的成本。卖家在对产品创新、提高产品竞争力的同时，也需要注意平衡成本的投入。

综上所述，关于补货和定价决策，商超还应收集顾客消费、供需分析、预期利润、外部影响因素如政策调整等和产品成本的相关数据。

六、 模型的评价、改进与推广

6.1 模型的优点

对数据进行初步的描述统计，实现数据的可视化，直观地了解数据的变化趋势与特点，为建立合理的模型提供了有效的经验基础。同时，建立的模型能与实际紧密结合，可根据不同情况对问题进行求解，使得问题的解决更加贴合实际，增强模型的通用性。

*k-means++*将不同单品的销售量进行分类处理，算法新颖，计算简便，便于理解，在处理大数据时，该算法可以实现较好的伸缩性和高效性。并且考虑到销售的不同情况，进行相关的数据清洗，去除掉离群值，得到的预测结果合理性较强。

根据原材料数据量的差异性，分别采用 *ARIMA* 时间序列预测模型和灰色预测模型，估计原料的批发价格。

基于“成本加成定价关系”得到收益模型，可通过各商品的利润率和销售量直接进行求解，便于理解，可操作性强。*SARIMA* 模型可以对存在季节性变化的时间序列进行建模，通过控制误差指标的变化，提高预测效果的稳定性。

6.2 模型的缺点

SARIMA 模型通过学习时间序列的短期相关性和季节性变化规律进行预测,无法准确判断未来的趋势变化。

模糊规划模型中的参数确立(模糊性)仍需要更专业的处理，才能拟合更为精确的利润率，实现更为直观的操作计算。同时，模型检验的方法较为复杂，实现起来比较繁琐。

6.3 模型的推广

基于模糊规划改进后得到的收益—利润系数模型，抛开以往对销售交易和原料成本的讨论，简化问题，得出每种单品不同销售量、利润率和收益的关系，直观性强，可用于其他复杂关系的模型中。

时间序列预测模型中，数据分布范围广、样本量多、运输过程麻烦，指标没有严格的限制。为样本数据的挖掘更有针对性，应用科学的方法建立更多指标，对为量化的指标建立相关关系。

七、 参考文献

- [1] 刘明. 经济时间序列的 ARIMA 类模型构建[J]. 统计与决策, 2014(08):29-32. DOI:10.13546/j.cnki.tjyjc.2014.08.006.
- [2] 赵肖肖, 朱宁, 黄黎平. 基于 ARIMA 模型的时间序列建模算法和实证分析[J]. 桂林电子科技大学学报, 2012, 32(05):410-415. DOI:10.16725/j.cnki.cn45-1351/tn.2012.05.004.
- [3] 郑秀国, 杨娟, 钱婷婷等. 近十年上海蔬菜价格的 SARIMA 分析与预测[J]. 上海农业学报, 2020, 36(06):138-141. DOI:10.15955/j.issn1000-3924.2020.06.25.
- [4] 杨皓旭. 蔬菜价格与销量的相关性分析研究——以油麦菜为例[J]. 食品安全导刊, 2018(21):3. DOI:CNKI:SUN:SPAQ.0.2018-21-149.
- [5] 刘保政, 刘德宝, 高立群. 供不应求季节性商品的价格控制和生产销售决策模型[J]. 东北大学学报: 自然科学版, 2005, 26(11):4. DOI:10.3321/j.issn:1005-3026.2005.11.006.
- [6] 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[D]. 中南林业科技大学, 2023. DOI:10.27662/d.cnki.gznlc.2022.000680.
- [7] 闻卉. 不同主体主导下生鲜农产品供应链的利润最大化策略研究[D]. 华中农业大学, 2021. DOI:10.27158/d.cnki.ghznu.2020.000036.
- [8] 曼昆 《经济学原理》

附录

问题一：

附录 1

问题一中各品类蔬菜的销售量变化曲线图

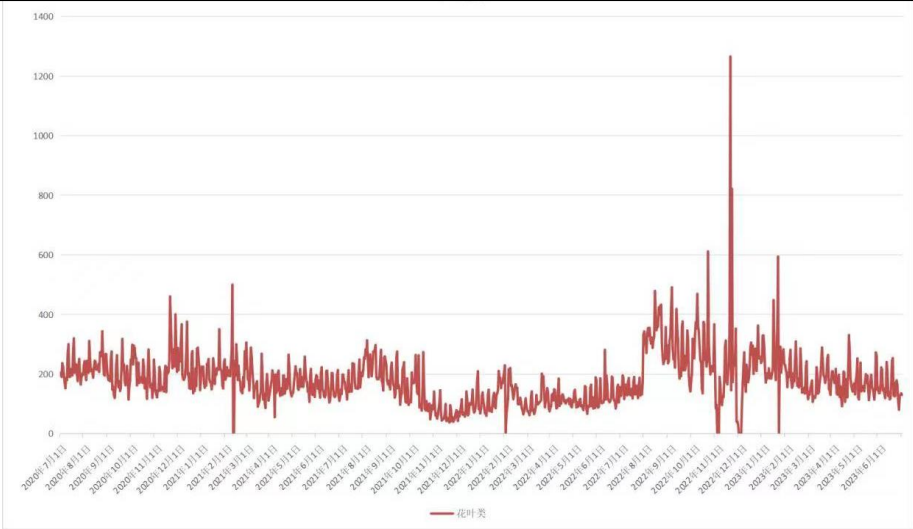


图 1 花叶类销售量变化曲线

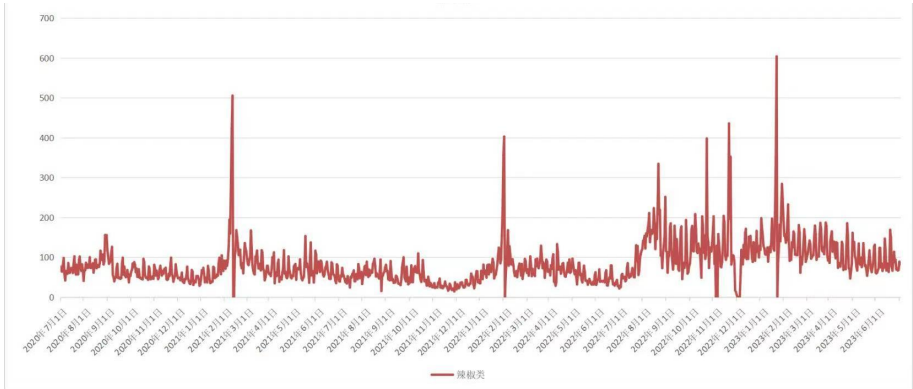


图 2 辣椒类销售量变化曲线

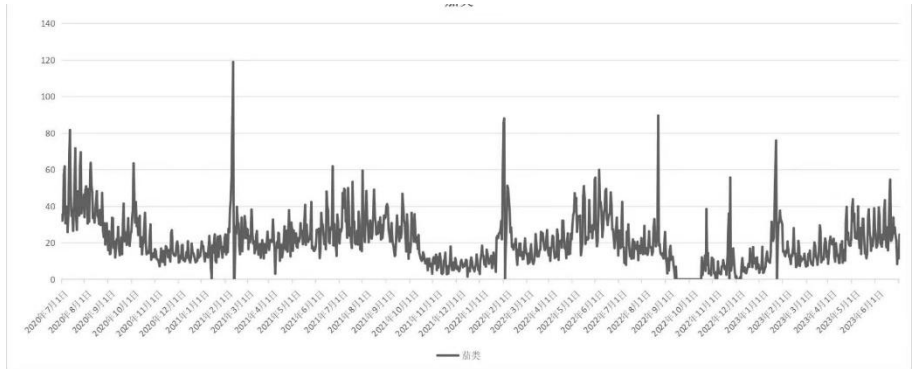


图 3 茄类销售量变化曲线

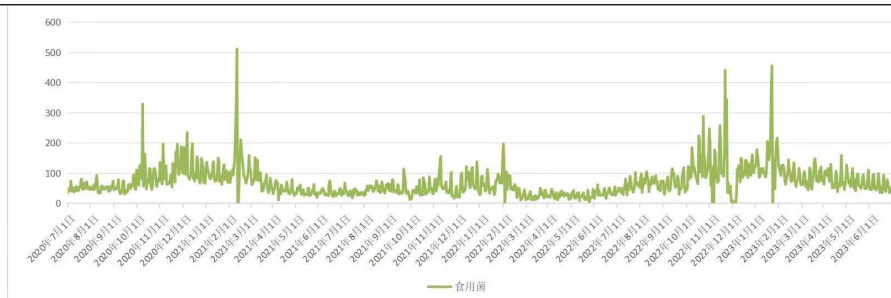


图 4 食用菌类销售变化曲线

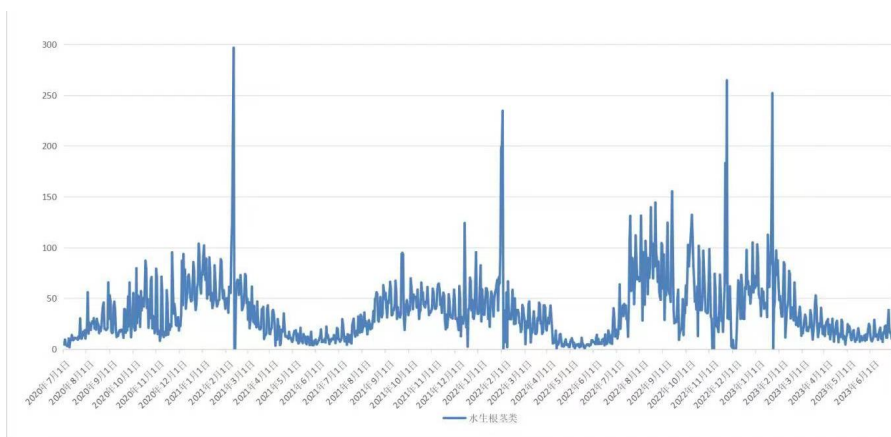


图 5 水生根茎类销售变化曲线

附录 2

数据预处理分类 Python 代码

```
import pandas as pd

# 读取 Excel 文件
df = pd.read_excel('水生根茎类.xlsx')

# 将销售日期列从字符串转换为日期时间对象
df['销售日期'] = pd.to_datetime(df['销售日期'], format='%Y 年%m 月%d 日')

result = df.groupby(['销售日期'])['销量(千克)'].sum().reset_index()
result.head()

unique_combinations = df.groupby(['销售日期'])['交易额'].sum().reset_index()
unique_combinations.head()

# 合并结果 DataFrame 和包含销售单价的 DataFrame
result = pd.merge(result, unique_combinations, on=['销售日期'], how='left')
result.head()

result.to_excel('水生根茎类_销量单价拟合.xlsx', index=False)
```


附录 3

聚类结果

单品编码	单品名称	销售天数	每日销售量	聚类结果
102900005115168	牛首生菜	102	8.821931373	4
102900005115199	四川红香椿	151	2.206774834	3
102900005115250	西峡花菇(1)	297	9.463690236	3
102900005115625	本地小毛白菜	47	2.574893617	4
102900005115748	白菜苔	92	7.811695652	4
102900005115762	苋菜	635	8.031592126	2
102900005115779	云南生菜	810	19.64254444	1
102900005115786	竹叶菜	667	10.85571814	1
102900005115793	小白菜	478	3.571763598	2
102900005115816	南瓜尖	4	4.607	4
102900005115823	上海青	919	8.277210011	1
102900005115854	萝卜叶	164	2.532469512	3
102900005115861	牛首油菜	285	13.46163509	3
102900005115878	茼蒿	514	7.996424125	2
102900005115885	蔡甸藜蒿	303	3.100907591	3
102900005115908	菜心	711	6.324496484	1
102900005115946	木耳菜	421	3.721235154	2
102900005115960	大白菜	480	39.97337083	2
102900005115977	豌豆尖	39	2.180717949	4
102900005115984	云南油麦菜	782	13.17821483	1
102900005116219	红尖椒	314	1.396792994	3
102900005116226	青尖椒	617	1.92306483	2
102900005116233	红椒(1)	755	4.579966887	1
102900005116257	紫茄子(2)	1022	13.30919863	1
102900005116509	青茄子(1)	845	4.161849704	1
102900005116530	西峡香菇(1)	821	14.51915591	1
102900005116547	金针菇(1)	448	10.48542411	2
102900005116639	马齿苋	93	1.014139785	4
102900005116714	西兰花	1076	25.59221933	1
102900005116790	黄心菜(1)	447	6.512973154	2
102900005116806	黑油菜	27	3.74537037	4
102900005116837	平菇	362	7.024748619	2
102900005116899	净藕(1)	1057	25.6853737	1
102900005116905	青杭椒(1)	5	0.3634	4
102900005116912	姬菇(1)	308	1.218253247	3
102900005116943	红杭椒	588	1.12	2
102900005117056	泡泡椒(精品)	228	42.55756579	3

102900005117209	七彩椒(1)	284	0.926732394	3
102900005117353	白玉菇(1)	8	2.75	4
102900005118572	本地上海青	61	9.781918033	4
102900005118817	菠菜	730	7.145836986	1
102900005118824	高瓜(1)	544	2.168477941	2
102900005118831	娃娃菜	688	13.05523256	1
102900005119098	双孢菇	224	2.8023125	3
102900005119104	蟹味菇(1)	51	2	4
102900005119944	海鲜菇(1)	326	1.540383436	3
102900005119968	青杭椒(2)	90	1.427777778	4
102900005119975	红薯尖	567	9.099529101	2
102900005122654	枝江红菜苔	384	12.92305729	2
102900005123880	灯笼椒(1)	217	0.977156682	3
102900005125808	小米椒	585	2.474107692	2
102900005125815	杏鲍菇(1)	452	5.320745575	2
102900005128748	黄花菜	6	0.297333333	4
102900011000175	快菜	10	9.1272	4
102900011000328	螺丝椒	858	9.081796037	1
102900011000335	紫圆茄	8	0.95725	4
102900011000571	随州泡泡青	142	4.755077465	3
102900011000632	紫白菜(1)	15	0.8834	4
102900011000861	红线椒	32	0.86603125	4
102900011001219	红灯笼椒(1)	272	1.024308824	3
102900011001561	莲蓬(个)	86	24.36046512	4
102900011001691	菱角	105	1.204019048	4
102900011001806	姬菇(包)	359	2.983286908	2
102900011001813	金针菇(袋)(1)	123	17.47154472	4
102900011002414	田七	96	0.471375	4
102900011006689	东门口小白菜	435	3.548225287	2
102900011006948	外地茼蒿	269	2.723475836	3
102900011006955	冰草	17	0.961	4
102900011007044	银耳(朵)	141	2.035460993	3
102900011007464	紫苏	84	0.359190476	4
102900011007471	薄荷叶	31	0.244129032	4
102900011007495	蒲公英	18	0.235111111	4
102900011007969	红莲藕带	96	1.046552083	4
102900011008133	丝瓜尖	19	0.748789474	4
102900011008164	奶白菜	762	7.632950131	1
102900011008485	芥菜	14	3.695357143	4
102900011008492	大芥兰	2	3.74	4
102900011008515	面条菜	33	0.392	4

102900011008522	甜白菜	340	13.78017941	3
102900011008577	猴头菇	3	1	4
102900011008676	荠菜	14	1.134785714	4
102900011009246	鲜木耳(1)	252	0.841948413	3
102900011009277	荸荠	213	7.847061033	3
102900011009444	大龙茄子	197	5.924964467	3
102900011009772	水果辣椒(橙色)	1	0.415	4
102900011009970	青梗散花	444	18.90492342	2
102900011010563	鸡枞菌	3	0.405	4
102900011010891	野藕(1)	164	2.73695122	3
102900011011058	黑牛肝菌	3	0.361333333	4
102900011011546	海鲜菇(袋)(1)	172	7.360465116	3
102900011011669	秀珍菇	17	0.865176471	4
102900011012482	茶树菇(袋)	48	1.875	4
102900011012871	黑皮鸡枞菌	16	0.3621875	4
102900011012994	杏鲍菇(袋)	327	4.107033639	3
102900011013274	白玉菇(袋)	698	4.193409742	1
102900011015384	马兰头	14	0.242142857	4
102900011015391	甘蓝叶	1	0.943	4
102900011016701	芜湖青椒(1)	858	32.82556061	1
102900011016909	花茄子	81	1.251592593	4
102900011018095	虫草花(袋)	141	6.936170213	3
102900011018132	野生粉藕	23	2.209347826	4
102900011021644	洪山菜苔	11	2.272727273	4
102900011021675	赤松茸	3	0.796	4
102900011021699	牛排菇	10	0.6602	4
102900011021842	洪湖莲藕(粉藕)	393	15.39949109	2
102900011022030	组合椒系列	176	3.642045455	3
102900011022764	长线茄	454	5.498707048	2
102900011022849	本地黄心油菜	115	11.95758261	4
102900011022924	青菜苔	3	10.09166667	4
102900011023075	活体银耳	1	1	4
102900011023464	小青菜(1)	492	6.642262195	2
102900011023976	净藕(2)	8	7.777625	4
102900011024010	鲜藕带(袋)	7	3	4
102900011026502	鲜粽子叶	8	0.79	4
102900011026618	艾蒿	4	2.628	4
102900011026793	杏鲍菇(250 克)	5	2	4
102900011027462	奶白菜苗	4	4.20725	4
102900011027479	小皱皮	81	2.020246914	4
102900011027615	菊花油菜	5	1.6944	4

102900011028407	余干椒	20	1.69535	4
102900011029176	辣妹子	73	1.638972603	4
102900011029275	紫尖椒	2	0.877	4
102900011029299	紫螺丝椒	10	0.6843	4
102900011029305	水果辣椒	30	0.952366667	4
102900011029688	双沟白菜	2	1.443	4
102900011030042	枝江红菜苔(份)	52	6.019230769	4
102900011030059	云南生菜(份)	400	35.8125	2
102900011030097	云南油麦菜(份)	410	21.5804878	2
102900011030103	茼蒿(份)	83	7.072289157	4
102900011030110	菠菜(份)	386	16.43005181	2
102900011030134	菜心(份)	275	7.465454545	3
102900011030141	上海青(份)	287	10.69686411	3
102900011030158	小青菜(2)	46	6.434782609	4
102900011030400	洪山菜薹珍品手提袋	1	3	4
102900011030417	洪山菜薹莲藕拼装礼盒	3	1	4
102900011030561	赤松茸(盒)	5	1	4
102900011030608	牛排菇(盒)	25	1.36	4
102900011030615	猪肚菇(盒)	6	1.166666667	4
102900011030622	黑牛肝菌(盒)	3	1.333333333	4
102900011030639	黑皮鸡枞菌(盒)	3	1.333333333	4
102900011030905	鱼腥草(份)	130	3.615384615	4
102900011030912	海鲜菇(份)	162	8.030864198	3
102900011030929	鲜木耳(份)	182	4.071428571	3
102900011031100	小米椒(份)	488	22.19877049	2
102900011031216	冰草(盒)	23	1.260869565	4
102900011031582	青尖椒(份)	193	4.829015544	3
102900011031599	杏鲍菇(份)	19	5.368421053	4
102900011031735	七彩椒(份)	10	1.8	4
102900011031742	灯笼椒(份)	4	1.5	4
102900011031759	红灯笼椒(份)	8	1.625	4
102900011031841	双孢菇(份)	4	2.5	4
102900011031858	金针菇(份)	1	1	4
102900011031926	虫草花(份)	245	4.526530612	3
102900011031995	姬菇(份)	100	6.82	4
102900011032022	小皱皮(份)	441	11.73469388	2
102900011032114	荸荠(份)	6	1.666666667	4
102900011032176	紫苏(份)	33	1.212121212	4
102900011032206	青杭椒(份)	110	4.154545455	4
102900011032213	红杭椒(份)	192	3.682291667	3
102900011032220	水果辣椒(份)	85	3.329411765	4

102900011032237	青线椒(份)	280	8.239285714	3
102900011032244	红尖椒(份)	246	3.87398374	3
102900011032251	螺丝椒(份)	382	21.55759162	2
102900011032282	襄甜红菜苔(袋)	7	5.857142857	4
102900011032343	七彩椒(2)	258	1.434236434	3
102900011032350	灯笼椒(2)	101	0.758376238	4
102900011032367	红灯笼椒(2)	182	1.180274725	3
102900011032480	白蒿	14	0.158857143	4
102900011032589	鱼腥草	31	0.922612903	4
102900011032619	西峡香菇(2)	69	5.997695652	4
102900011032626	鲜木耳(2)	41	0.69597561	4
102900011032633	海鲜菇(2)	43	1.206860465	4
102900011032640	姬菇(2)	33	0.513727273	4
102900011032732	高瓜(2)	159	2.828201258	3
102900011032787	竹叶菜(份)	78	13.79487179	4
102900011032848	姜蒜小米椒组合装(小份)	366	5.103825137	2
102900011033081	春菜	22	1.136181818	4
102900011033173	槐花	20	0.465	4
102900011033234	红薯尖(份)	58	12.53448276	4
102900011033241	蔡甸藜蒿(份)	16	4.0625	4
102900011033531	红珊瑚(粗叶)	1	0.682	4
102900011033562	红橡叶	1	0.419	4
102900011033586	绿牛油	1	1.153	4
102900011033906	保康高山大白菜	200	32.42368	3
102900011033913	紫白菜(2)	2	0.3075	4
102900011033920	黄心菜(2)	223	8.44290583	3
102900011033937	金针菇(2)	89	5.318146067	4
102900011033944	杏鲍菇(2)	293	4.452395904	3
102900011033968	西峡花菇(2)	10	0.924	4
102900011033975	青茄子(2)	52	3.663884615	4
102900011033982	紫茄子(1)	44	6.753590909	4
102900011033999	圆茄子(1)	3	1.660333333	4
102900011034026	枝江青梗散花	337	17.27469139	3
102900011034200	苋菜(份)	56	7.607142857	4
102900011034217	小白菜(份)	47	7.255319149	4
102900011034224	小青菜(份)	179	22.66480447	3
102900011034231	奶白菜(份)	244	28.4057377	3
102900011034262	红椒(份)	27	3.444444444	4
102900011034316	芝麻苋菜	1	9.381	4
102900011034323	鲜粽叶	9	0.667666667	4
102900011034330	双孢菇(盒)	293	14.4334471	3

102900011034354	鲜粽叶(袋)(1)	7	28.57142857	4
102900011034439	青红杭椒组合装(份)	228	5.513157895	3
102900011034538	花菇(一人份)	3	1	4
102900011034569	洪湖莲藕(脆藕)	12	3.333333333	4
102900011034705	菌菇火锅套餐(份)	4	2	4
102900011035078	红椒(2)	288	2.670861111	3
102900011035481	外地茼蒿(份)	22	5.772727273	4
102900011035511	净藕(3)	70	4.982857143	4
102900011035740	蟹味菇与白玉菇双拼(盒)	137	2.708029197	3
102900011035764	龙牙菜	63	2.965571429	4
102900011035771	黄白菜(1)	70	2.221371429	4
102900011035788	西峡香菇(份)	65	5.830769231	4
102900011035849	大白菜秧	4	4.1345	4
102900011035962	野藕(2)	6	1.899666667	4
102900011036068	菌蔬四宝(份)	4	3.25	4
102900011036242	青红尖椒组合装(份)	24	1.875	4
102900011036266	鹿茸菇(盒)	2	1.5	4
102900011036686	木耳菜(份)	25	1.08	4
102900051000463	圆茄子(2)	501	2.091195609	2
102900051000890	芥兰	1	0.671	4
102900051000944	洪湖藕带	258	2.889744186	3
102900051004294	青线椒	747	2.971594378	1
102900051006229	藕尖	3	6.204666667	4
102900051009220	油菜苔	18	2.262555556	4
102900051009336	虫草花	82	0.48752439	4
102900051010455	黄白菜(2)	904	8.836272124	1
102900051010790	紫贝菜	53	0.509849057	4
106930274220092	蟹味菇(袋)	100	3.99	4
106930274620090	海鲜菇(袋)(2)	51	13.88235294	4
106931885000035	绣球菌	6	1.166666667	4
106931885000356	绣球菌(袋)	2	1	4
106949711300068	金针菇(袋)(3)	258	9.879844961	3
106949711300167	金针菇(袋)(2)	215	14.76744186	3
106949711300259	金针菇(盒)	429	36.35431235	2
106956146480197	白玉菇(2)	59	4.254237288	4
106956146480203	蟹味菇(2)	32	2.96875	4
106957634300010	白玉菇(盒)	42	4	4
106957634300058	蟹味菇(盒)	32	2.53125	4
106958851400125	海鲜菇(袋)(4)	314	6.843949045	3
106971533450003	海鲜菇(包)	119	10.94957983	4
106971533455008	海鲜菇(袋)(3)	240	7.395833333	3

106971563780002	鲜粽叶(袋)(2)	4	2	4
106972776821582	鲜粽叶(袋)(3)	31	4.129032258	4
106973223300667	虫草花(盒)(2)	1	1	4
106973990980123	和丰阳光海鲜菇(包)	19	5.736842105	4

附录 4

聚类 MATLAB 代码

```
% 执行 K-means 聚类分析
X ;

% 指定聚类数量为 4
K = 4;

% 使用 kmeans 函数对数据 X 进行聚类分析,将聚类结果存储在 idx 和聚类中心
存储在 C 中
[idx, C] = kmeans(X, K);

% 显示聚类结果
disp('聚类结果:');
for i = 1:K
    fprintf('聚类 %d:\n', i);

    % 从聚类结果中提取属于第 i 类的数据
    clusterData = X(idx == i, :);

    % 显示该类的数据
    disp(clusterData);
end

% 显示聚类中心
disp('聚类中心:');
disp(C);
```

聚类绘图 Python 代码

```
import pandas as pd
import matplotlib.pyplot as plt

# 读取 Excel 文件
excel_file = 'cluster_result.xlsx' # 替换成您的 Excel 文件路径
data = pd.read_excel(excel_file)

# 分割数据为不同分类
categories = data['聚类结果'].unique()
```

```
# 设置不同分类的颜色
colors = ['r', 'g', 'b', 'c'] # 替换成您希望使用的颜色

# 创建一个散点图
plt.figure(figsize=(10, 6)) # 设置图的大小
for category, color in zip(categories, colors):
    category_data = data[data['聚类结果'] == category]
    plt.scatter(category_data['销售天数'], category_data['每日销售量'],
label=f'Classification {category}', c=color)

# 添加标签和图例
plt.xlabel('Days')
plt.ylabel('Sales')
plt.legend()

# 显示图
plt.title('Scatter plot of clustering results')
plt.show()
```

问题二

附录 1

SARIMA 模型最佳参数

花菜类	Best Parameters: ((1, 1, 1), (0, 1, 1, 7))
花叶类	Best Parameters: ((1, 0, 1), (0, 1, 1, 7))
茄类	Best Parameters: ((1, 1, 1), (0, 1, 1, 7))
辣椒类	Best Parameters: ((1, 1, 1), (0, 1, 1, 7))
食用菌	Best Parameters: ((1, 1, 1), (1, 1, 1, 7))
水生根茎类	Best Parameters: ((1, 1, 1), (0, 1, 1, 7))

附录 2

SARIMA 预测模型 Python 代码

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from statsmodels.tsa.statespace.sarimax import SARIMAX

# 这里假设"data"的列表, 包含了你提供的所有数据点
df = pd.read_excel('sales_summary.xlsx')

data = df.iloc[:, 1].tolist()

# 将数据转换为 pandas 的 Series 对象, 设置日期索引 (假设从某一天开始)
start_date = pd.to_datetime('2020-07-01')
date_index = pd.date_range(start=start_date, periods=len(data), freq='D')
time_series = pd.Series(data, index=date_index)
time_series.head()

plt.figure(figsize=(12, 6))
plt.plot(time_series)
plt.title("Time Series Data")
plt.xlabel("Date")
plt.ylabel("Value")
plt.show()

# SARIMA 模型的参数需要根据数据的性质来选择, 这里使用了一些默认值, 你可以根据需要进行调整
order = (1, 1, 1) # ARIMA 部分的阶数
seasonal_order = (0, 1, 1, 7) # 季节性部分的阶数和季节周期 (一天)

sarima_model = SARIMAX(time_series, order=order, seasonal_order=seasonal_order)
sarima_fit = sarima_model.fit(dispatch=False)
```

```

forecast_days = 7
forecast = sarima_fit.get_forecast(steps=forecast_days)

# 获取预测值和置信区间
forecast_mean = forecast.predicted_mean
forecast_ci = forecast.conf_int()

# 打印预测值
print(forecast_mean)

plt.figure(figsize=(12, 6))
plt.plot(time_series, label='Observed')
plt.plot(forecast_mean, label='Forecast', color='red')
plt.fill_between(forecast_ci.index, forecast_ci.iloc[:, 0], forecast_ci.iloc[:, 1],
color='pink')
plt.title("SARIMA Forecast")
plt.xlabel("Date")
plt.ylabel("Value")
plt.legend()
plt.show()

# 寻找最佳参数
from itertools import product
import statsmodels.api as sm

# 定义一组候选参数
p = d = q = range(0, 2)
pdq = list(product(p, d, q))
seasonal_pdq = [(x[0], x[1], x[2], s) for x in pdq for s in [7]] # 假设季节性周期为 7

best_aic = np.inf
best_params = None

# 通过循环来搜索最佳参数组合
for param in pdq:
    for param_seasonal in seasonal_pdq:
        try:
            mod = sm.tsa.statespace.SARIMAX(time_series,
                                             order=param,
                                             seasonal_order=param_seasonal,
                                             enforce_stationarity=False,
                                             enforce_invertibility=False)

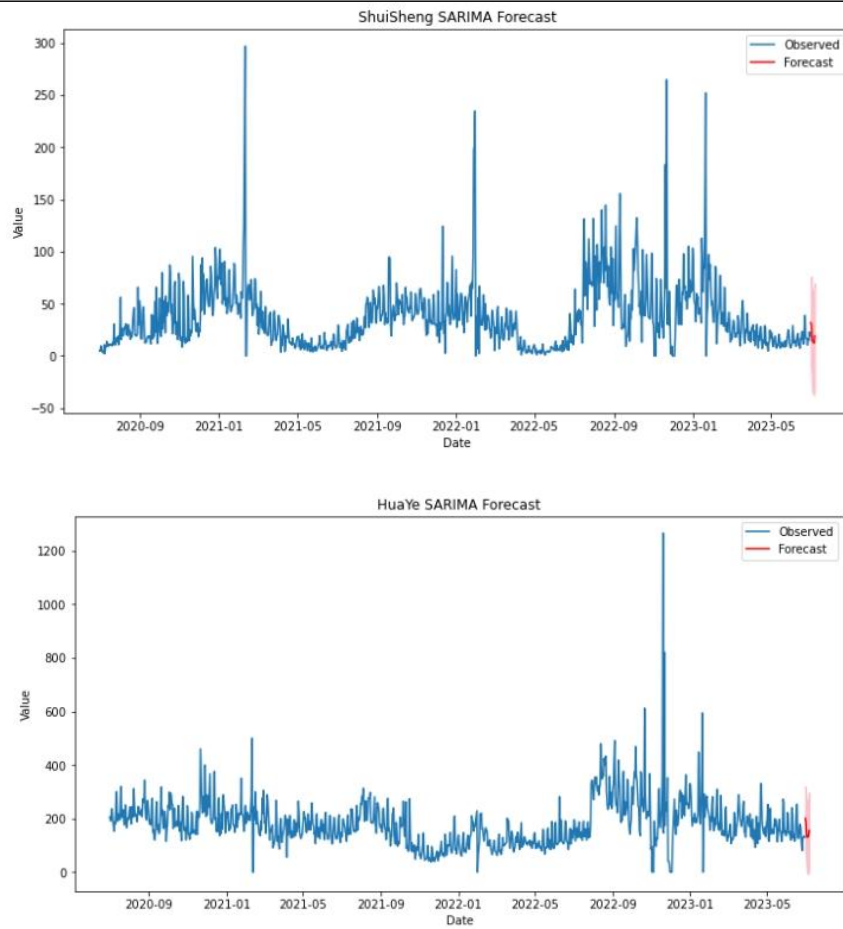
            results = mod.fit()
            if results.aic < best_aic:
                best_aic = results.aic
                best_params = (param, param_seasonal)
        except:
            continue

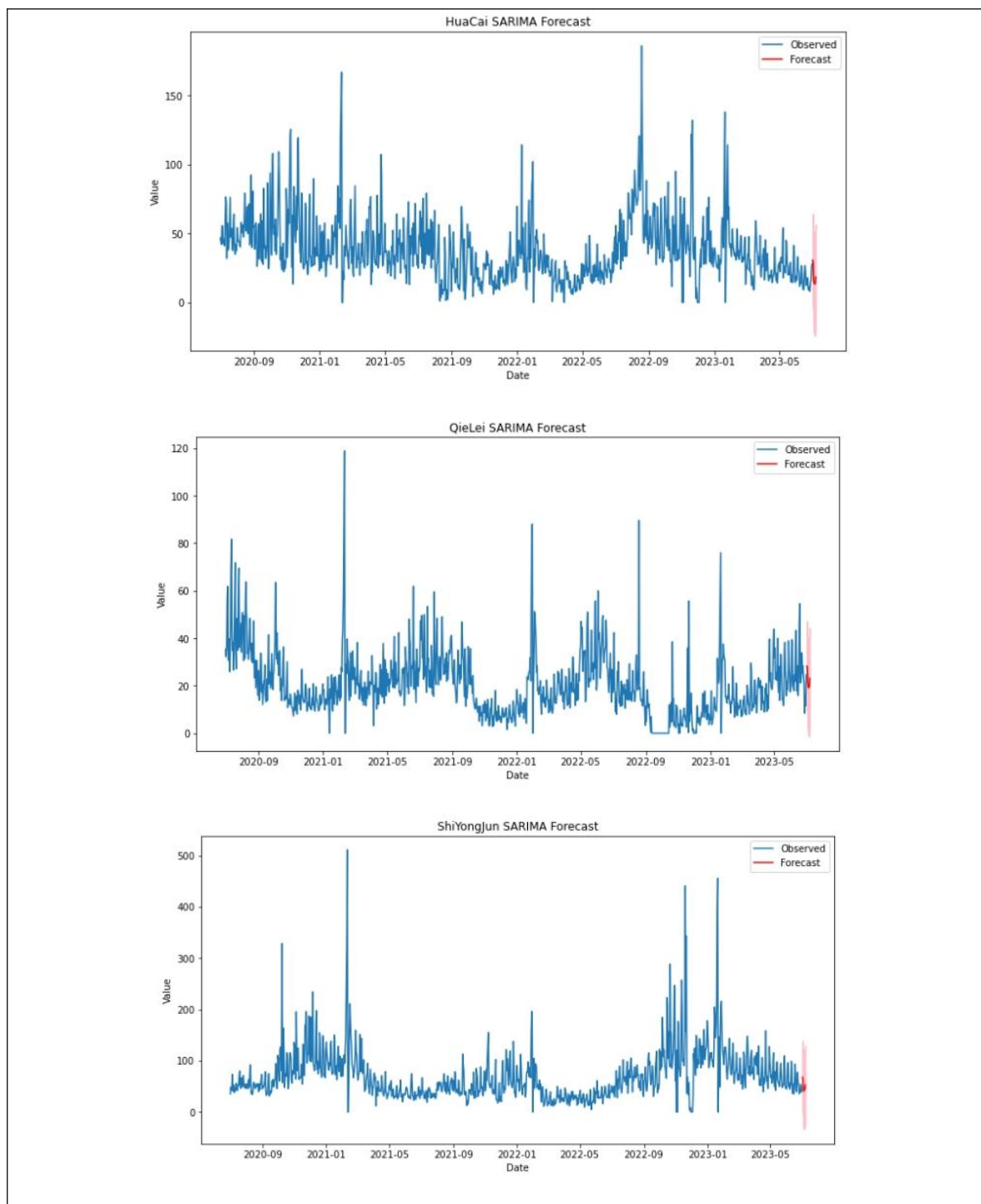
```

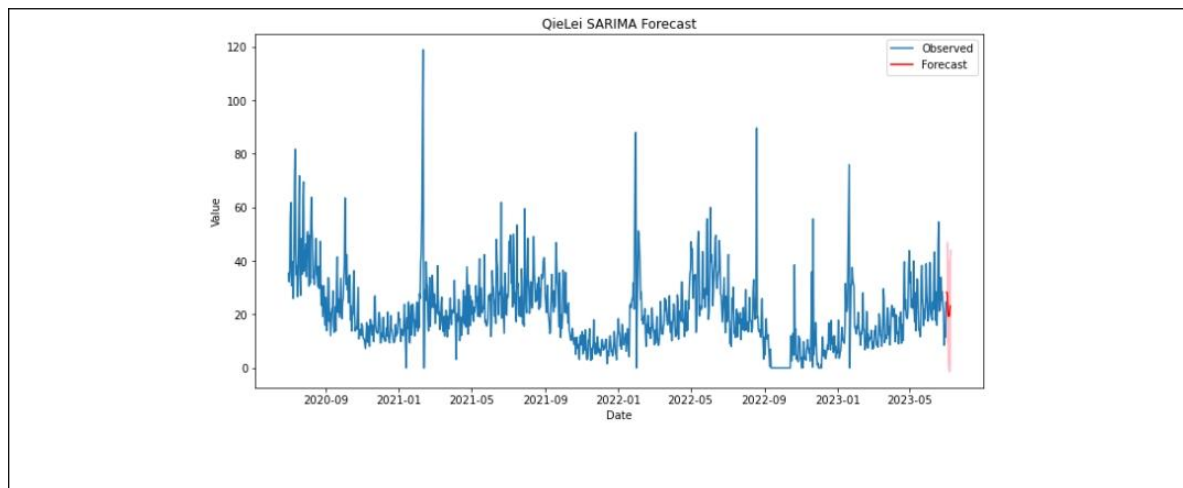
```
# 输出最佳参数
print("Best AIC:", best_aic)
print("Best Parameters:", best_params)
```

附录 3

SARIMA 模型预测图







附录 4

销量和定价关系拟合 Python 代码

```
import pandas as pd
```

```
# 读取 Excel 文件
```

```
df = pd.read_excel('茄类_销量单价拟合.xlsx')
```

```
df.info
```

```
from scipy.optimize import curve_fit
```

```
import numpy as np
```

```
col2_data = df.iloc[:, 1].values
```

```
col4_data = df.iloc[:, 3].values
```

```
def inverse_proportion(x, a):
```

```
    return a / x
```

```
params, covariance = curve_fit(inverse_proportion, col2_data, col4_data)
```

```
a = params[0]
```

```
print("拟合参数 a =", a)
```

灰色预测 Matlab 代码

主函数

```
%% 数据预处理和可视化
```

```
% 清空工作区和命令窗口
```

```
clear;clc
```

```

day =[1:1:19]'; % 横坐标表示年份，写成列向量的形式（加'就表示转置）
x0 =
[14.98 ,14.99 ,14.98 ,14.99 ,14.99 ,16.56 ,16.59 ,16.48 ,16.32 ,16.32
,16.46 ,18
.75 ,18.76 ,18.56 ,28.36 ,28.36 ,28.36 ,28.37 ,32.49 ]'; %原始数据
序列，写成列向量的形式（加'就表示转置）

% 绘制原始数据的时间序列图
figure(1); % 指定图形编号
plot(day,x0,'o-'); grid on; % 绘制原始数据的时间序列图
set(gca,'xtick',day(1:1:end)) % 设置 x 轴横坐标的间隔为 1
xlabel('天数 day(天)'); ylabel('销售数量 amount（单位 kg）'); % 添加坐标轴标签

%% 数据准备和模型选择

ERROR = 0; % 建立一个错误指标，一旦出错就指定为 1

% 检查是否有负数元素
if sum(x0<0) > 0
disp('warning: 时间序列中存在不符合要求的负数数据！请重新修改序列数据！')
ERROR = 1;
end

% 检查数据量是否太少
n = length(x0);
disp(strcat('该数据集长度为:',num2str(n)))
if n<=3
disp('warning: 数据量太小，无法满足预测要求！')
ERROR = 1;
end

% 对于大数据集，建议考虑其他方法
if n>10
disp('warning: 数据量较大，请考虑使用其他方法！')
end

% 如果输入的是行向量，则转置为列向量
if size(x0,1) == 1
x0 = x0';
end
if size(day,1) == 1
day = day';
end

```

```

%% 准指数规律检验

if ERROR == 0
disp('_____')
disp('准指数规律检验: ')
x1 = cumsum(x0);
rho = x0(2:end) ./ x1(1:end-1) ;
% 绘制光滑度图, 标出临界值
figure(2)
plot(day(2:end),rho,'o-',[day(2),day(end)],[0.5,0.5],'-'); grid
on;
text(day(end-1)+0.2,0.55,'临界线')
set(gca,'xtick',day(2:1:end))
xlabel('年份 year (年)'); ylabel('原始数据的光滑度');
disp(strcat('指标 1: 光滑比小于 0.5 的数据占比为',num2str(100*sum(rho<0.5)/(n-1)),'%'))
disp(strcat('指标 2: 前两个时期外, 光滑比小于 0.5 的数据占比为',num2str(100*sum(rho(3:end)<0.5)/(n-3)),'%'))
disp('standard: 指标 1 一般要大于 0.6, 指标 2 要大于 0.9, 您认为本例数据通过目标检验吗? ')
Judge = input('您认为可以通过准指数规律的检验吗? 可以的话通过请输入 1, 不能请输入 0: ');
if Judge == 0
disp('warning: 灰色预测模型不适合该数据,请重新进行模型的选择! ')
ERROR = 1;
end
disp('_____')
end

%% 模型选择和预测

if ERROR == 0
if n > 4
disp('因为原数据的期数大于 4, 所以我们可以将数据组分为训练组和试验组')
if n > 7
test_num = 3;
else
test_num = 2;
end
train_x0 = x0(1:end-test_num);
disp('训练数据是: ')
disp(mat2str(train_x0'))
test_x0 = x0(end-test_num+1:end);
disp('试验数据是: ')
disp(mat2str(test_x0'))
disp('-----')

```

```

-')
% 使用三种模型对训练数据进行训练，返回的 result 就是往后预测 test_num 期
的数据
disp(' ')
disp('***下面是传统的 GM(1,1)模型预测的详细过程***')
result1 = gm11(train_x0, test_num);
disp(' ')
disp('***下面是进行新信息的 GM(1,1)模型预测的详细过程***')
result2 = new_gm11(train_x0, test_num);
disp(' ')
disp('***下面是进行新陈代谢的 GM(1,1)模型预测的详细过程***')
result3 = metabolism_gm11(train_x0, test_num);
% 比较三种模型对试验数据的预测结果
disp(' ')
disp('-----')
-')
test_year = day(end-test_num+1:end);
figure(3)
plot(test_year,test_x0,'o-',test_year,result1,'*- ',test_year,result2,'+- ',test_year,result3,'x- '); grid on;
set(gca,'xtick',day(end-test_num+1): 1 :day(end))
legend('试验组的真实数据','传统 GM(1,1)预测结果','新信息 GM(1,1)预测结果','新陈代谢 GM(1,1)预测结果')
xlabel('年份'); ylabel('销售总量');
SSE1 = sum((test_x0-result1).^2);
SSE2 = sum((test_x0-result2).^2);
SSE3 = sum((test_x0-result3).^2);
disp(strcat('传统 GM(1,1)对于试验组预测的误差平方和为',num2str(SSE1)))
disp(strcat('新信息 GM(1,1)对于试验组预测的误差平方和为',num2str(SSE2)))
disp(strcat('新陈代谢 GM(1,1)对于试验组预测的误差平方和为',num2str(SSE3)))
if SSE1<SSE2
if SSE1<SSE3
choose = 1;
else
choose = 3;
end
elseif SSE2<SSE3
choose = 2;
else
choose = 3;
end
Model = {'传统 GM(1,1)模型','新信息 GM(1,1)模型','新陈代谢 GM(1,1)模型'};
disp(strcat('因为',Model(choose),'的误差平方和最小,所以我们应该选择其

```

```

进行预测'))
disp('-----
-')
%% 选择最佳模型进行预测
predict_num = input('请输入您要往后预测的期数: ');
[result, x0_hat, relative_residuals, eta] = gm11(x0, predict_num);
if choose == 2
result = new_gm11(x0, predict_num);
end
if choose == 3
result = metabolism_gm11(x0, predict_num);
end
%% 输出使用最佳模型预测的结果
disp('-----
-')
disp('对原始数据的拟合结果: ')
for i = 1:n
disp(strcat(num2str(day(i)), ' : ', num2str(x0_hat(i))))
end
disp(strcat('往后预测', num2str(predict_num), '期的结果: '))
for i = 1:predict_num
disp(strcat(num2str(day(end)+i), ' : ', num2str(result(i))))
end
else
disp('因为数据只有 4 期，所以我们直接将三种方法的结果求平均即可~')
predict_num = input('请输入您要往后预测的期数: ');
disp(' ')
disp('***下面是传统的 GM(1,1)模型预测的详细过程***')
[result1, x0_hat, relative_residuals, eta] = gm11(x0, predict_num);
disp(' ')
disp('***下面是进行新信息的 GM(1,1)模型预测的详细过程***')
result2 = new_gm11(x0, predict_num);
disp(' ')
disp('***下面是进行新陈代谢的 GM(1,1)模型预测的详细过程***')
result3 = metabolism_gm11(x0, predict_num);
result = (result1+result2+result3)/3;
disp('对原始数据的拟合结果: ')
for i = 1:n
disp(strcat(num2str(day(i)), ' : ', num2str(x0_hat(i))))
end
disp(strcat('传统 GM(1,1)往后预测', num2str(predict_num), '期的结果: '))
for i = 1:predict_num
disp(strcat(num2str(day(end)+i), ' : ', num2str(result1(i))))
end
disp(strcat('新信息 GM(1,1)往后预测', num2str(predict_num), '期的结果: '))

```

```

'))
for i = 1:predict_num
disp(strcat(num2str(day(end)+i), ' : ', num2str(result2(i))))
end
disp(strcat('新陈代谢 GM(1,1) 往后预测', num2str(predict_num), ' 期的结果: '))
for i = 1:predict_num
disp(strcat(num2str(day(end)+i), ' : ', num2str(result3(i))))
end
disp(strcat('三种方法求平均得到的往后预测', num2str(predict_num), ' 期的结果: '))
for i = 1:predict_num
disp(strcat(num2str(day(end)+i), ' : ', num2str(result(i))))
end
end
%% 绘制相对残差和级比偏差的图形
figure(4)
subplot(2,1,1)
plot(day(2:end), relative_residuals, '*-'); grid on;
legend('相对残差'); xlabel('年份');
set(gca, 'xtick', day(2:1:end))
subplot(2,1,2)
plot(day(2:end), eta, 'o-'); grid on;
legend('级比偏差'); xlabel('年份');
set(gca, 'xtick', day(2:1:end))
disp(' ')
disp('****下面将输出对原数据拟合的评价结果****')
%% 残差检验
average_relative_residuals = mean(relative_residuals);
disp(strcat('平均相对残差为', num2str(average_relative_residuals)))
if average_relative_residuals < 0.1
disp('残差检验的结果表明: 该模型对原数据的拟合程度非常不错')
elseif average_relative_residuals < 0.2
disp('残差检验的结果表明: 该模型对原数据的拟合程度达到一般要求')
else
disp('残差检验的结果表明: 该模型对原数据的拟合程度不太好, 建议使用其他模型预测')
end
%% 级比偏差检验
average_eta = mean(eta);
disp(strcat('平均级比偏差为', num2str(average_eta)))
if average_eta < 0.1
disp('级比偏差检验的结果表明: 该模型对原数据的拟合程度非常不错')
elseif average_eta < 0.2
disp('级比偏差检验的结果表明: 该模型对原数据的拟合程度达到一般要求')
else

```

```

disp('级比偏差检验的结果表明：该模型对原数据的拟合程度不太好，建议使用其他模型预测')
end
disp(' ')
disp('-----')
disp(' ')
%% 绘制最终的预测效果图
figure(5)
plot(day,x0,'-o', day,x0_hat,'-m',
day(end)+1:day(end)+predict_num,result,'-b'); grid on;
hold on;
plot([day(end),day(end)+1],[x0(end),result(1)],'-b')
legend('原始数据','拟合数据','预测数据')
set(gca,'xtick',[day(1):1:day(end)+predict_num])
xlabel('年份'); ylabel('销售数量');
end

GM(1,1)函数代码
function [result, x0_hat, relative_residuals, eta] = gm11(x0,
predict_num)

n = length(x0); % 数据的长度
x1=cumsum(x0); % 计算一次累加值
z1 = (x1(1:end-1) + x1(2:end)) / 2; % 计算紧邻均值生成数列（长度为 n-1）
% 将从第二项开始的 x0 当成 y，z1 当成 x，来进行一元回归 y = kx + b
y = x0(2:end); x = z1;
% 下面的表达式就是第四讲拟合里面的哦~ 但是要注意，此时的样本数应该是 n-1，少了一项哦
k =
((n-1)*sum(x.*y)-sum(x)*sum(y))/((n-1)*sum(x.*x)-sum(x)*sum(x));
b =
(sum(x.*x)*sum(y)-sum(x)*sum(x.*y))/((n-1)*sum(x.*x)-sum(x)*sum(x));
a = -k; %注意：k = -a 哦
% 注意： -a 就是发展系数，b 就是灰作用量
disp('现在进行 GM(1,1) 预测的原始数据是：')
disp(mat2str(x0')) % mat2str 可以将矩阵或者向量转换为字符串显示
disp(strcat('最小二乘法拟合得到的发展系数为',num2str(-a),'，灰作用量是',num2str(b)))
disp('*****分割线*****')
x0_hat=zeros(n,1); x0_hat(1)=x0(1); % x0_hat 向量用来存储对 x0 序列的拟合值，这里先进行初始化
for m = 1: n-1
x0_hat(m+1) = (1-exp(a))*(x0(1)-b/a)*exp(-a*m);
end
result = zeros(predict_num,1); % 初始化用来保存预测值的向量

```

```

for i = 1: predict_num
result(i) = (1-exp(a))*(x0(1)-b/a)*exp(-a*(n+i-1)); % 带入公式直接计算
end

% 计算绝对残差和相对残差
absolute_residuals = x0(2:end) - x0_hat(2:end); % 从第二项开始计算绝对残差，因为第一项是相同的
relative_residuals = abs(absolute_residuals) ./ x0(2:end); % 计算相对残差，注意分子要加绝对值，而且要使用点除
% 计算级比和级比偏差
class_ratio = x0(2:end) ./ x0(1:end-1) ; % 计算级比 sigma(k) = x0(k)/x0(k-1)
eta = abs(1-(1-0.5*a)/(1+0.5*a)*(1./class_ratio)); % 计算级比偏差
end

```

新信息模型

```

function [result] = new_gm11(x0, predict_num)
result = zeros(predict_num,1); % 初始化用来保存预测值的向量
for i = 1 : predict_num
result(i) = gm11(x0, 1); % 将预测一期的结果保存到 result 中
x0 = [x0; result(i)]; % 更新 x0 向量，此时 x0 多了新的预测信息
end
end

```

新城代谢模型

```

function [result] = metabolism_gm11(x0, predict_num)
result = zeros(predict_num,1); % 初始化用来保存预测值的向量
for i = 1 : predict_num
result(i) = gm11(x0, 1); % 将预测一期的结果保存到 result 中
x0 = [x0(2:end); result(i)]; % 更新 x0 向量，此时 x0 多了新的预测信息，并且删除了最开始的那个向量
end
end

```


问题三：

附录 1

规划 *MATLAB* 代码

%水生根茎类

```
c = [11.35 10.61 6.65 5.93 5.76 5.40 2.30]';  
Aeq = [1 1 1 1 1 1 1];  
beq = 19.958;  
lb = [2.5 2.5 2.5 2.5 2.5 2.5 2.5]';  
[x fval] = linprog(c, [], [], Aeq, beq, lb)
```

%食用菌类

```
c = [7.97 4.15 3.06 2.88 2.29]';  
Aeq = [1 1 1 1 1];  
beq = 48.148;  
lb = [2.5 2.5 2.5 2.5 2.5]';  
[x fval] = linprog(c, [], [], Aeq, beq, lb)
```

%茄类

```
c = [6.80 3.47 3.26 2.63]';  
Aeq = [1 1 1 1];  
beq = 23.936;  
lb = [2.5 2.5 2.5 2.5]';  
[x fval] = linprog(c, [], [], Aeq, beq, lb)
```

%辣椒类

```
c = [13.32 7.56 3.21 2.80]';  
Aeq = [1 1 1 1];  
beq = 85.383;  
lb = [2.5 2.5 2.5 2.5]';  
[x fval] = linprog(c, [], [], Aeq, beq, lb)
```

%花叶类

```
c = [7.52 4.12 3.33 3.22 3.21 2.91 2.58 2.51 2.37 2.31 2.22 ]';  
Aeq = [1 1 1 1 1 1 1 1 1 1 1];  
beq = 139.767;  
lb = [2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5]';  
[x fval] = linprog(c, [], [], Aeq, beq, lb)
```

%花菜类

```
c = [3.5 3.17]';  
Aeq = [1 1];  
beq = 22.097;  
lb = [2.5 2.5]';  
[x fval] = linprog(c, [], [], Aeq, beq, lb)
```

附录 2					
日补货量和定价策略					
编号	蔬菜名称	品类	补货量	不打折销售单价	打折销售单价
102900011032343	七彩椒(2)	辣椒类	77.88	25.32	16.49
102900011032732	高瓜(2)	水生根茎类	4.96	22.71	14.80
102900051000944	洪湖藕带	水生根茎类	2.50	29.66	17.38
102900005115250	西峡花菇(1)	食用菌	38.15	21.48	19.51
102900011035078	红椒(2)	辣椒类	2.50	17.53	11.64
102900011006948	外地茼蒿	花叶类	114.77	16.16	14.20
102900011033982	紫茄子(1)	茄类	16.44	11.04	6.83
102900011018132	野生粉藕	水生根茎类	2.50	14.99	0.00
102900011007969	红莲藕带	水生根茎类	2.50	10.70	0.00
102900005118824	高瓜(1)	水生根茎类	2.50	13.83	11.31
102900011001691	菱角	水生根茎类	2.50	14.57	6.73
102900011035740	蟹味菇与白玉菇双拼(盒)	食用菌	2.50	9.03	5.41
102900005118817	菠菜	花叶类	2.50	9.98	6.52
102900011034026	枝江青梗散花	花菜类	19.60	8.44	5.54
102900011022764	长线茄	茄类	2.50	10.76	6.00
102900005115779	云南生菜	花叶类	2.50	8.26	7.14
102900005116257	紫茄子(2)	茄类	2.50	8.76	6.58
102900005115946	木耳菜	花叶类	2.50	7.15	6.21
102900011000328	螺丝椒	辣椒类	2.50	10.68	8.96
102900005115823	上海青	花叶类	2.50	7.53	7.25
102900005116714	西兰花	花菜类	2.50	9.93	9.18
102900011013274	白玉菇(袋)	食用菌	2.50	7.10	4.93
102900005115908	菜心	花叶类	2.50	7.36	5.86
102900011031926	虫草花(份)	食用菌	2.50	5.65	3.60
102900011032848	姜蒜小米椒组合装(小份)	辣椒类	2.50	4.27	2.55
102900051000463	圆茄子(2)	茄类	2.50	7.59	4.00
102900005115984	云南油麦菜	花叶类	2.50	6.90	6.73
102900011023464	小青菜(1)	花叶类	2.50	5.89	5.87
102900011036686	木耳菜(份)	花叶类	2.50	4.34	2.50
102900011008164	奶白菜	花叶类	2.50	4.87	3.11
102900005116899	净藕(1)	水生根茎类	2.50	8.05	7.16
102900011034330	双孢菇(盒)	食用菌	2.50	5.77	3.65
102900011030097	云南油麦菜(份)	花叶类	2.50	4.39	2.86