蔬菜类商品自动定价与补货决策的优化模型

摘　要

鉴于附件提供的数据量极大，进行数据预处理和**量化分析**显得尤为必要。首先，对于扫码销售时间作出**饼图**，顾客大多数的采购时间在早上 9 点至 11 点和下午 15 点至 18 点，占比分别达到 33% 以及 37.3%。其次，分析这三年里商品价格的变化情况，并挑选出十个具有代表性的单品作出**箱线图**，发现黑皮鸡枞菌的销售单价是最高的。

针对问题一，首先，本文选取蔬菜品类的每个月的**总销售量**和**平均销售量**作为分析蔬菜品类分布规律的指标。然后，对蔬菜单品选取总销售量作为分析单品分布规律的指标。再通过 Excel 处理数据，得到**条形图**，经过描述性分析得知水生根茎类、茄类、和食用菌的销售量具有**季节性**，而花叶类、花菜类、辣椒类一年之内的销售量较为稳定。还有同一类之间的不同单品之间的销量差异也较大。并通过 Python 求解 Pearson 相关系数，得到六类蔬菜品类之间的相关性和所有蔬菜单品之间的相关性其中，在品类中，花叶类和花菜类的相关性最强；在单品中，云南生菜与云南油麦菜的相关性最强。

针对问题二，首先，本文对销量 (千克) 和销售单价 (元/千克) 进行一次 **Spearman 秩相关性**分析和 **Kendall's** $\tau - b$ **相关性**分析共同验证了六大品类的销售总量与成本加成定价的关系大致是**负相关**的。接着，因为商家需要在不知道具体商品和进货价格的情况下，决策当日的蔬菜补货，所以本文使用 2023 年 6 月份的数据建立了**时间序列预测模型**去预测未来 7 天的补货量，并且通过 2021 和 2022 年的 6 月份的数据去验证**周期性**。最后，构建约束条件，求解单目标规划问题获得了一周的**期望最大收益** (即 2023 年 7 月 1-7 日) 的近似值为 **8113 元**。

针对问题三，首先，本文对 2023 年 6 月 24-30 日的每一个单品进行销量总量的量化分析，发现这一周内一共销售了 49 种单品。根据平均最小陈列量，本文初步筛选了 29 种高销量的商品。然后，拟合了**回归预测方程**，并通过了统计检验，初步预测了 7 月 1 日的单品补货量。最后，对初步预测的单品补货量添加**扰动范围**和考虑筛选 27、28 或 29 种单品建立了**双目标规划模型**，通过 Python 编程求解获得了该天的期望最大收益的近似值为 **1359 元**。

针对问题四，首先，本文中要提高补货和定价决策的稳定性，则需要从减少蔬菜商品的损耗率以及使商品的价格更加合理两个方面进行考虑，可结合生活实际经验和收集到的一些数据，我们得出收集各类蔬菜商品每一天的**库存数量**、当地的**天气预报数据**、**消费者反馈数据**、**附近商超的相关数据**这些数据都可以使得商超更好地制定蔬菜商品的补货和定价决策，从而提高商超的收益。

关键词：量化分析　时间序列预测　回归预测　Spearman 秩相关性分析　双目标规划模型

一、 问题重述

1.1 问题背景

生鲜商超中，蔬菜类商品因保鲜期较短，品相也容易变差，很多在一天内未被售出的商品第二天就不适合再次出售。商超每天会基于历史销售和需求情况进行补货。但是，因为进货的时间是在凌晨，而具体的蔬菜品类和进货价格事先是未知的，所以商家需要做出一个预估来决定当天的补货策略。一般来说，定价是基于“成本加成定价”方法，并且对于运损和品相变差的商品会进行打折销售。市场需求分析在这里是非常重要的，尤其是蔬菜的销售量与时间存在关联性。此外，蔬菜供应品种在 4 月至 10 月相对丰富，因此如何选择合适的销售组合是关键。

1.2 问题重述

题目要求我们综合附件所给数据建立合理的数学模型并解决以下问题：

问题一： 问题一要求我们根据附件 1 和附件 2 的单品及品类信息和单品历史销售数据，分析蔬菜的各品类及单品销售量的分布规律，并研究各品类及单品之间可能存在的关联关系。

问题二： 问题二要求我们根据附件提供的数据，以品类为单位，分析各蔬菜品类的销售总量与成本加成定价的关系，并为 2023 年 7 月 1-7 日制定各蔬菜品类的日补货总量和定价策略，使得商超收益最大。

问题三： 考虑到商超销售空间有限，要求可售单品数量控制在 27-33 个，且每个单品的订购量需满足最小陈列量 2.5 千克。要求我们基于 2023 年 6 月 24-30 日的可售品种，提供 7 月 1 日的单品补货量和定价策略，在满足市场对各品类蔬菜商品需求的前提下，使得商超收益最大。

问题四： 为了更好地制定蔬菜商品的补货和定价策略，题目要求我们思考商超还需要采集哪些相关数据，并解释这些数据如何帮助解决上述问题。

二、 问题分析

2.1 问题一的分析

附件 1 提供了品类和单品的编码和名称，附件 2 提供了商超近三年销售流水明细数据。本文通过 Python 以单品编码为关键字对两个附件进行合并。问题一要求分析出蔬菜各品类及单品销售量的分布规律及相互关系。通过 Python 和 Excel 处理数据，得到六个品类蔬菜和每个单品的每月总销售量和平均销售量以及所有蔬菜商品的销售量排名进行分析品类和单品的销量分布规律。最后，通过 Python 求解皮尔斯相关系数，得到六类蔬菜品类之间的相关性以及所有蔬菜单品之间的相关性。再根据具体情况进行具体分析，深挖底层原因。

2.2 问题二的分析

本文对销量 (千克) 和销售单价 (元/千克) 进行一次 Spearman 秩相关性分析和 Kendall's $\tau - b$ 相关性分析去验证六大品类的销售总量与成本加成定价的关系大致是负相关的。接着，因为商家需要在不知道具体商品和进货价格的情况下，决策当日的蔬菜补货，所以本文使用 2023 年 6 月份的数据建立了时间序列预测模型去预测未来 7 天的补货量，并且通过 2021 和 2022 年的 6 月份的数据去验证周期性。最后，构建约束条件，求解单目标规划问题获得了一周的期望最大收益 (即 2023 年 7 月 1-7 日) 的近似值。

2.3 问题三的分析

本文对 2023 年 6 月 24-30 日的每一个单品进行销量总量的量化分析，确定这一周内一共销售的单品的种数。根据平均最小陈列量，本文初步筛选出高销量的商品。然后，拟合回归预测方程，并通过统计检验，初步预测 7 月 1 日的单品补货量。最后，对初步预测的单品补货量添加扰动范围和考虑筛选 27、28 或 29 种单品建立了双目标规划模型，通过 Python 编程求解获得该天的期望最大收益的近似值。

2.4 问题四的分析

问题四是指在已经提供的 4 个附件数据的基础上，还能收集哪些有效数据，可以使得商超更好地制定蔬菜商品的补货和定价决策，提高商超的盈利。首先，减少蔬菜商品的损耗率，并且使商品的价格更加合理，可以提高蔬菜商品销售量，由此提高商超的收益。其次，通过我们的生活经验和收集到的一些数据，我们得出收集各类蔬菜商品每一天的库存数量、当地的天气预报数据、消费者反馈数据、附近商超的相关数据这些数据都可以使得商超更好地制定蔬菜商品的补货和定价决策。

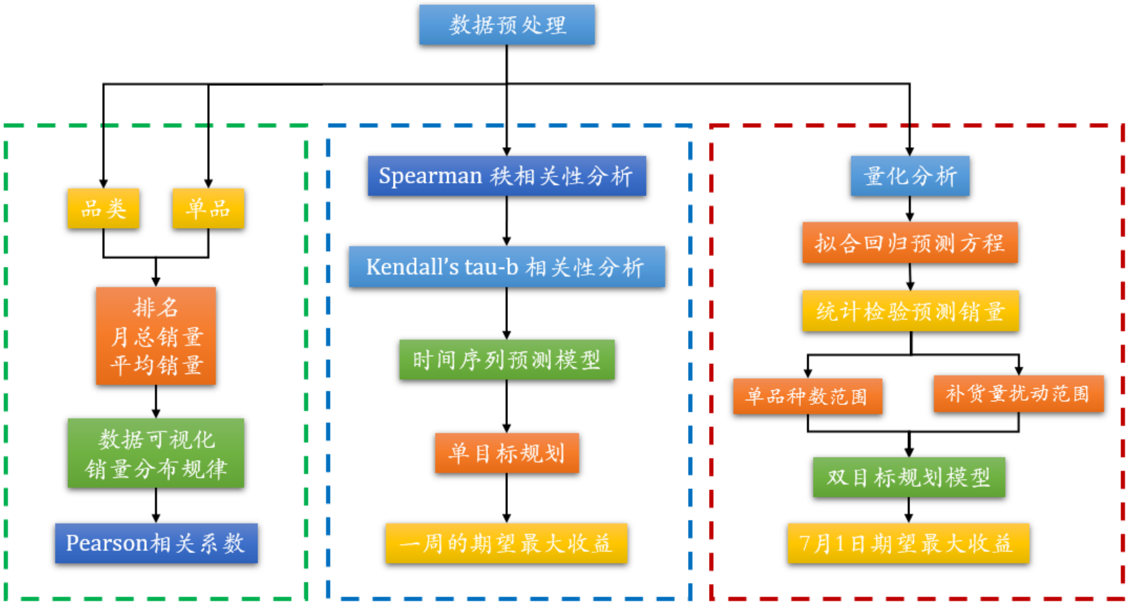
数据预处理
品类
单品
排名
月总销量
平均销量
数据可视化
销量分布规律
Pearson相关系数
Spearman 秩相关性分析
Kendall's tau-b 相关性分析
时间序列预测模型
单目标规划
一周的期望最大收益
量化分析
拟合回归预测方程
统计检验预测销量
单品种数范围
补货量扰动范围
双目标规划模型
7月1日期望最大收益

图 2-1: 模型流程图

三、 基本假设与符号说明

3.1 基本假设

1. 假设附件数据没有经过修改，真实有效；
2. 假设损坏率对应的商品可以全部打折售出；
3. 假设商超的现金流充足，而且没有仓库存储数量的限制；
4. 假设被预测的未来一周销售情况为全部当天售出的理想情况；
5. 假设每种单品的退货率相同且为固定值，以历史销售数据作为参考；
6. 假设不存在突发的自然灾害或人为灾难致使未来供货紧缺或生鲜产品被人们哄抢的特殊情况。

3.2 符号说明

符 号	含 义	单 位
S_i	售价	元/kg
W_i	批发价	元/kg
i	品类	\
j	单品	\
d_i	折扣价	元/kg
df_j	折扣系数	\
l_j	损耗率	\
r_j	补货量	kg
SS_j	销售空间	\
m_{jt}	单品加成率	\
m_i	品类的平均利润率	\

四、 数据预处理

本题提供了四个附件的数据。其中，附件 1 提供了 6 个蔬菜品类的商品信息，附件 2 提供了销售流水明细数据，附件 3 提供了蔬菜类商品的批发价格，附件 4 提供了蔬菜类商品的近期损耗率。观察数据后，得知了数据量比较庞大，所以需要对这些数据进行必要的预处理，如果数据出错，可以及时修改与调整，保证数据的健康性，提高后续计算结果的准确性。

经过 Python 编程遍历四个附件，本文发现这些附件的数据都比较健全，没有什么明显缺失或错误。由于附件 2 的数据更加庞大，指标也比较多。在这里，我们先考虑扫码销售时间、销售类型和是否打折销售。

首先，对于扫码销售时间，本文想要探究一下顾客在哪些时间段采购蔬菜较多。使用 Python 可视化处理，得到如下图表：

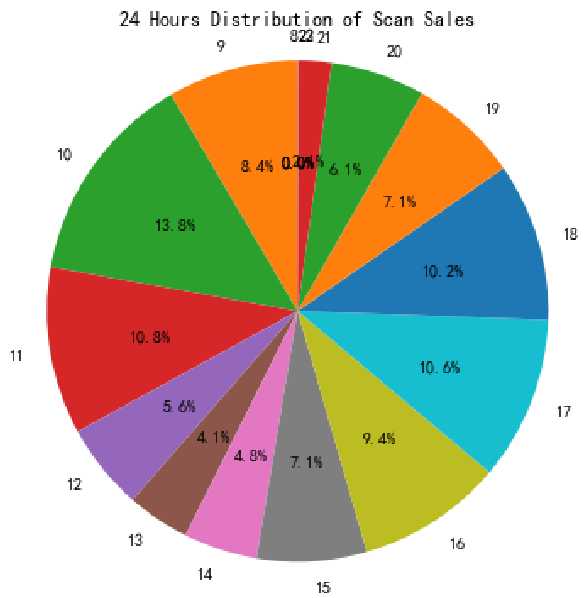

图 4-2: 顾客采购时间可视化

本文分析，顾客大多数的采购时间在早上 9 点至 11 点和下午 15 点至 18 点，占比分别达到 33% 以及 37.3%。根据现实生活中的常识，这两个时间段是比较合理的。同时也说明了数据的可靠性。

然后，本文对销售类型进行统计。销售类型有两种，分别是销售和退货。销售次数有 878042，退货次数有 461，总数是 878503。其中，销售占比达到 99.95%，退货占比达到 0.05%。退货率非常低，顾客满意度较高，说明了这家生鲜商超经营得不错。本文对是否打折销售进行统计。不打折销售的次数有 831137，打折销售的次数有 47366。不打折销售的占比达到 94.61%，打折销售的占比达到 5.39%。同时，本文注意到题目商超对运损和品相变差

的商品通常进行打折销售。直观说明了运输过程中保存得不错。

接着，本文在这里还想要去分析一下三年里商品价格的变化情况，所以使用 Python 编程得到最有价值的十个单品的价格变化的箱线图。

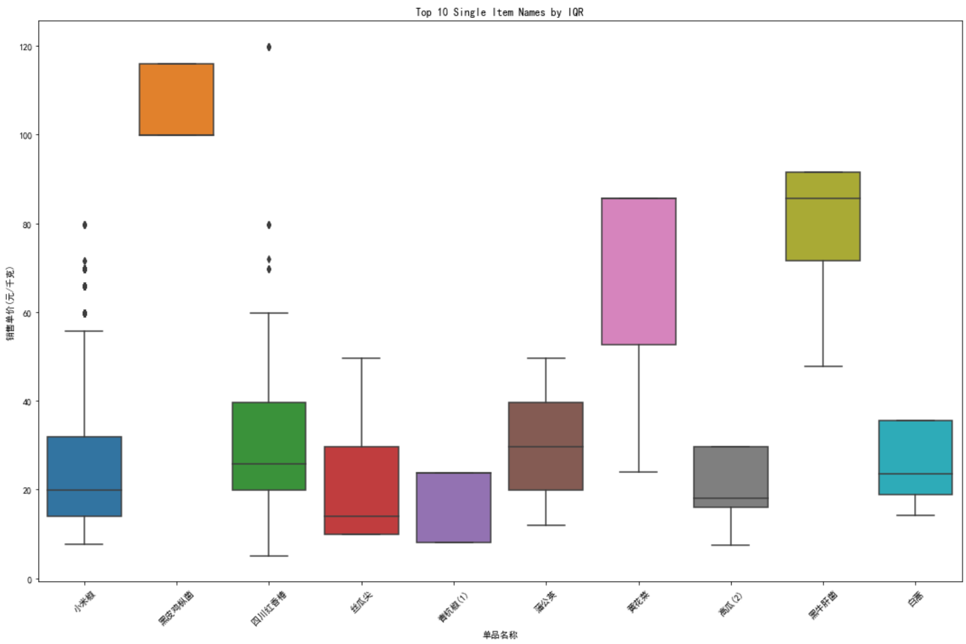

图 4-3: 最有价值的十个单品的价格变化可视化

最后，本文对附件 2 的销量 (千克) 进行正态性检验，结果显示不符合正态性检验。此外，本文还将同一天的销量 (千克) 进行求和，再次进行正态性检验，结果发现还是不符合正态性检验。同时，本文对销售单价 (元/千克) 也进行了正态性检验，发现依然不符合正态性检验。

后续还有根据每一问得需要进行一定数据预处理。

五、 问题一的模型建立与求解

5.1 品类的规律

问题一中需要分析蔬菜各品类及单品销售量的分布规律及相互关系(数据处理，附件一中有 251 种商品的单品编码和分类名称，我们通过 python 进行数据预处理，将附件一的数据和附件二的数据通过商品的单品编码使得分类名称和单品名称进行合并，使得预处理后的附件二数据中增加了商品的名称和商品的分类名称)，商品的销售量与时间往往存在一定的关联关系，这与蔬菜商品在不同时间的供应量一般是不同的以及不同季节很可能影响到消费者选择消费的不同蔬菜，正如冬吃萝卜夏吃姜的谚语所说的，所以，时间划分的处理的影响我们数据的有效性，由此我们将六类商品分类将 2020 年 7 月 1 日至 2023 年 6 月 30 日各商品的销售流水明细按月份进行统计分析，分析六类中每类的月平均销售量，总销售量和标准差。按月划分销售是由于，一个月内同一种蔬菜商品的销售量一般不会发生过大的变化。

通过 excel 分析预处理后的数据，我们得到了六类蔬菜类型 2020 年 7 月 1 日-2023 年 6 月 30 日每个月的总销售量和平均销售量的图像。通过这六张图像，我们分析六大类的商品得出 2022 年的商品销售量数量对比 2021、2022 和 2023 这三年的销售数据都有较大的变化，这很有可能是与 2022 年国内的疫情相关，较严重的疫情发生时使得人们不方便到商超中去购买商品，而选择从网上进行购买，那么会导致商超的蔬菜商品销售量减少，所以导致 2022 的商超中商品的销售量有较大的影响，由此，我们在考虑和分析问题中，对于 2022 年的销售数量我们分析的权重相较于其他年份会较少。

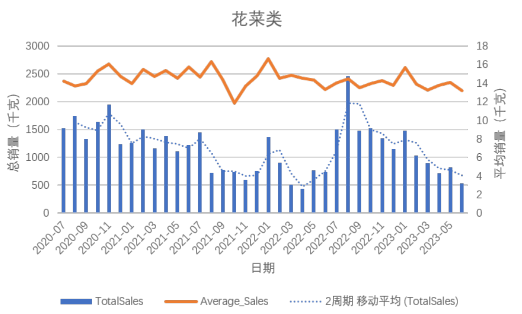

图 5-4: 花菜类

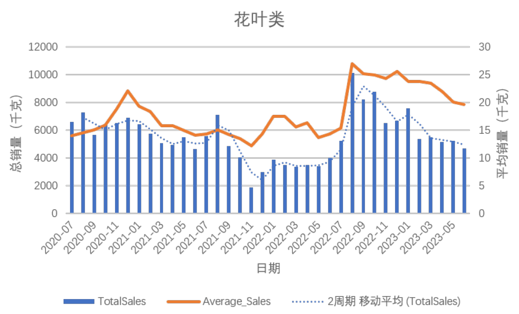

图 5-5: 花叶类

对于花菜类，本文分析这一类的蔬菜时发现这一类蔬菜每个月的销售量的波动幅度都较小，每天的平均销售数量都是在 12kg-16kg 的区间。这是因为目前很多地区都是采用大棚来种植花菜，所以它的栽培形式多样，秋冬季，夏秋季以及春季都有栽培的，还有高山栽培方式，因此已经形成了周年

供应，可根据市场需求上市。所以花菜类蔬菜一年四季都会成熟，没有太明显的生产季节性。

对于花叶类，这一类中，本文分析数据可以发现除了个别月份特殊，同一年大部分的月份的销售量都是在变化较少的。这说明花叶类也是不受季节性的影响的。在全年中的供应量都是比较充足，并且按年份来统计和分析那么花叶类的总销售量和平均销售量是增加的趋势。

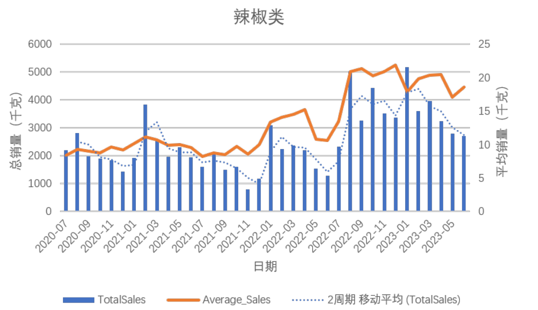

图 5-6: 辣椒类

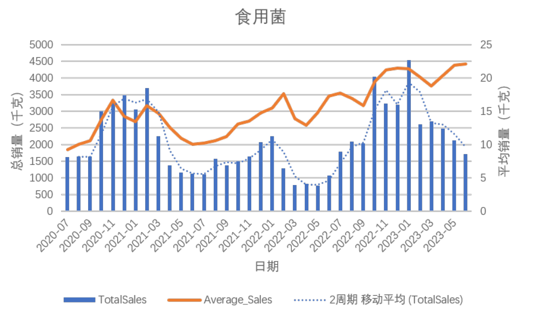

图 5-7: 食用菌

对于辣椒类，它的销售量在 2020-7 月到 2021-1 月之间每个月平均销售量和每个月的总销售量都是在一个较稳定的区间。在后面的几个月中以增加的趋势的增加后，每个月的平均销售量和总销售量又处于一个较稳定的区间。但是按年份来统计，辣椒类的销售量是增加的趋势。这可能是因为现在喜欢吃辣椒的人增加了。

对于茄类，本文分析得出每年的 7 月份都是一年中总销售量达到最高点的月份。此后销售量逐渐下降，11 月份左右的销售量是每一年销售量的最小数值，这一类的销售量同一年中不同月份之间的差值比是较大的，这说明茄类是夏季蔬菜，这时候市场茄类蔬菜供应量较多，那么销售价格较低并且这时候的茄类更加美味，由此销售量会大大增加。

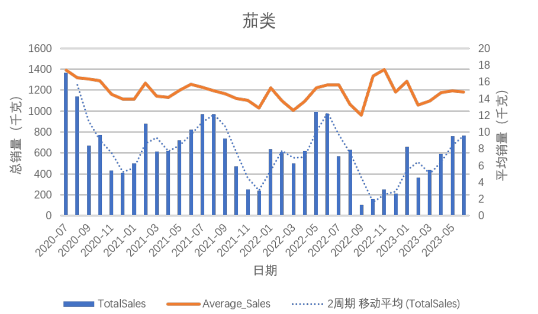

图 5-8: 茄类

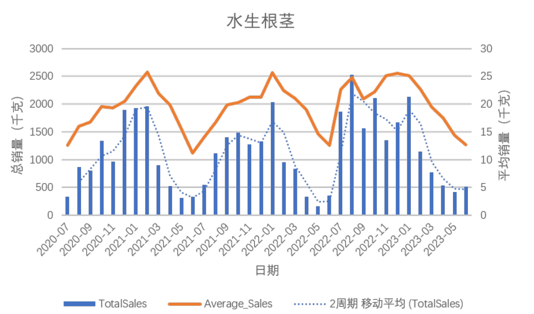

图 5-9: 水生根茎类

对于食用菌，秋季和冬季的销售量较大，是因为食用菌这一类的大部

分品种的生产的都是秋季和冬季，商品的单价相较于产生量少的季节价格更低，那么买的人也就更多一些。并且食用菌类销售量的总的趋势是逐年增加的。

对于水生根茎类，通过分析它的每个月的数据，可以发现水生根茎类是受季节影响较大的一类商品。夏季和秋季的总销售量是大于春季和冬季的总销售量。每年 5 月左右的销售量是全年销售量的最低点，此后销售量逐渐增加，1 月是全年中销售量最高的月份。并且水生根茎类是一种月份之间销售量增加或者减少幅度都较大的一类品种。

5.2 单品销售量的分布规律

通过 Python 编程进行数据预处理得到所有蔬菜单品 2020-7 到 2023-6 月三年时间内的总销售量排名。第一名是芜湖青椒，它的总销量是 28164kg, 最后一名是水果辣椒，都是辣椒类，这体现同一类之间的不同单品之间的销量差异也是较大的。

表 5-1: 排名前 50 的单品的总销售量

排名	单品名称	销量 (kg)	排名	单品名称	销量 (kg)
1	芜湖青椒 (1)	28164.331	26	奶白菜	5816.308
2	西兰花	27537.228	27	菠菜	5216.461
3	净藕 (1)	27149.44	28	小皱皮 (份)	5175
4	大白菜	19187.218	29	红薯尖	5159.433
5	云南生菜	15910.461	30	苋菜	5100.061
6	金针菇 (盒)	15596	31	枝江红菜苔	4962.454
7	云南生菜 (份)	14325	32	金针菇 (1)	4697.47
8	紫茄子 (2)	13602.001	33	甜白菜	4685.261
9	西峡香菇 (1)	11920.227	34	菜心	4496.717
10	小米椒 (份)	10833	35	双孢菇 (盒)	4229
11	云南油麦菜	10305.364	36	茼蒿	4110.162
12	泡泡椒 (精品)	9703.125	37	小青菜 (份)	4057
13	娃娃菜	8982	38	牛首油菜	3836.566
14	云南油麦菜 (份)	8848	39	青茄子 (1)	3516.763
15	青梗散花	8393.786	40	红椒 (1)	3457.875
16	螺丝椒 (份)	8235	41	小青菜 (1)	3267.993
17	黄白菜 (2)	7987.99	42	金针菇 (袋)(2)	3175
18	螺丝椒	7792.181	43	上海青 (份)	3070

19	上海青	7606.756	44	白玉菇 (袋)	2927
20	竹叶菜	7240.764	45	黄心菜 (1)	2911.299
21	奶白菜 (份)	6931	46	西峡花菇 (1)	2810.716
22	保康高山大白菜	6484.736	47	金针菇 (袋)(3)	2549
23	菠菜 (份)	6342	48	平菇	2542.959
24	洪湖莲藕 (粉藕)	6052	49	长线茄	2496.413
25	枝江青梗散花	5821.571	50	杏鲍菇 (1)	2404.977

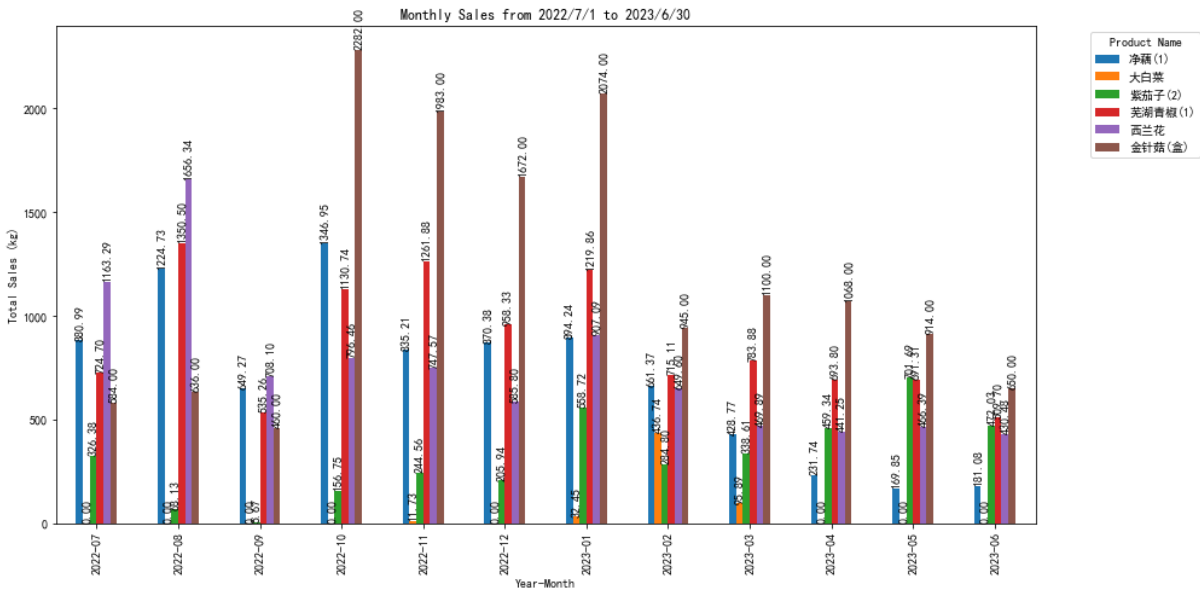

图 5-10: 单品月销量

如表 5-1 所示，可以看到销售量排名前 50 的单品，同时我们发现总销量排名前 10 中覆盖了六类蔬菜品类，由此我们选取芜湖青椒、西兰花、净藕、大白菜、金针菇、紫茄子六种品类来分析单品之间的相关性。将这六个单品 2022-7 月到 2023-6 月的数据通过 Python 编程处理，得到如图 5-9 六个品类的这一年月销总量的对比，我们发现金针菇的销量在冬季的远远高于其他季节，受季节影响较大。芜湖青椒和西兰花两种单品的销量则是每个月之间的变化量较少，维持在一个相对稳定的区间内。紫茄子这个单品是春冬季节的销量更好。与之相反的是净藕这个蔬菜单品则是夏秋季节销量更高。而大白菜在这一年的销量在六的单品中是最少的。

5.3 求解结果展示

将文件数据通过 Python 进行预处理, 得到了六大类蔬菜的三年 36 个月内的每个月历史总销售量。我们将水生根茎类、花叶类、花菜类、茄类、辣

椒类和食用菌分别用 x_1、x_2、x_3、x_4、x_5、x_6 来代表。再通过 python 进行皮尔逊相关系数模型的求解。得到如图的六类品类之间的相关性热力图。

其中，相关系数与相关程度一般划分为 0.8 - 1.0 极强相关，0.6 - 0.8 强相关，0.4 - 0.6 中等程度相关，0.2 - 0.4 弱相关，0 - 0.2 极弱相关或无相关

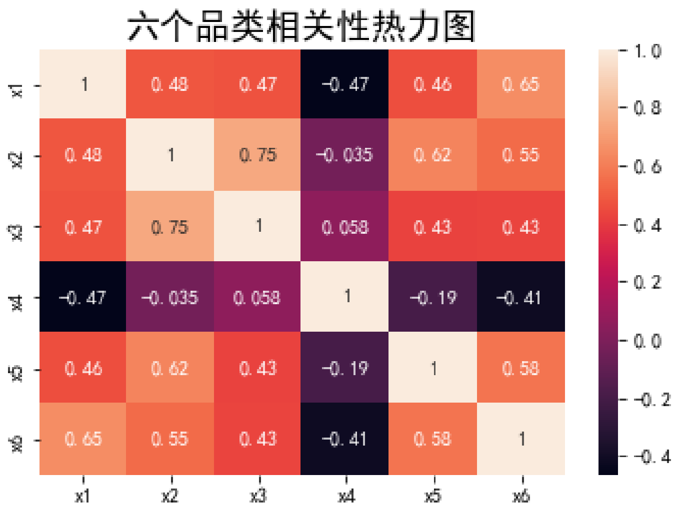

图 5-11: 品类相关性热力图

根据热力图可以得出花叶类、花菜类之间的相关是最强的达到 0.75，同时水生根茎类与食用菌以及花叶类与辣椒类的相关性也是比较强的。水生根茎类与茄类之间的相关性是最小的为-0.035.

再通过 Python 分析所有蔬菜单品之前的相关性，由于数据过多，导致热力图不能够清晰的进行分析。因此将所有蔬菜单品之间的相关性矩阵到出 Excel 中进行分析。由于蔬菜单品的数量较多，所有我们选取一些比较特殊的相关性进行分析，我们发现云南生菜与云南油麦菜之间的相关性为 0.98，这可能是因为云南生菜与云南油麦菜不仅仅来源于同一个产地，同时这个蔬菜单品的生产月份也是比较相近的，都是当季蔬菜，所有之间的销售量变化较相同，杏鲍菇与七彩椒的相关性为 0.9986，这可能是因为这两种蔬菜的做法就是炒杏鲍菇喜欢加入七彩椒一起制作。螺丝椒与小米椒的相关性为-0.71522，芜湖青椒与泡椒的相关性为-0.71255，这是因为人们大概率买入一种辣椒后不会再买入另一种辣椒，所有具有辣椒之间具有很强的负相关性。同时也有相关性非常弱的情况，例如冰菜与丝瓜尖之间的相关性为-0.0007 和净藕与上海青之间的相关性为 0.010205，这说明这些蔬菜商品的销售量之间没有什么影响。

六、 问题二的模型建立与求解

6.1 数据预处理

根据题目要求，首先需要分析各蔬菜品类的销售总量与成本加成定价的关系。本文对附件数据进行处理，得到每一品类的日期、在某一天的总销量 (千克) 和平均销售单价 (元/千克)。其中，本文根据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的要求，在日期中选取了 2023 年 6 月 1 日到 2023 年 6 月 30 日的数据。依次得到六个品类的数据。

6.2 Kendall's $\tau - b$ 相关性分析品类销量与成本加成定价的关系

经过数据预处理后，本文得到了六个品类的总销量 (千克) 和平均销售单价 (元/千克) 的数据。本文先对附件 2 全部的销量 (千克) 和销售单价 (元/千克) 进行一次 Spearman 秩相关性分析。Spearman 秩相关是一种非参数检验，它不要求数据是正态分布的。它衡量的是两个变量之间的单调关系。具体来说，它基于数据的秩来计算相关性，而不是数据的原始值。使用 Python 编程求解，得到：

表 6-2: Spearman 秩相关性

Spearman's rank correlation coefficient	-0.546
P-value	0.000

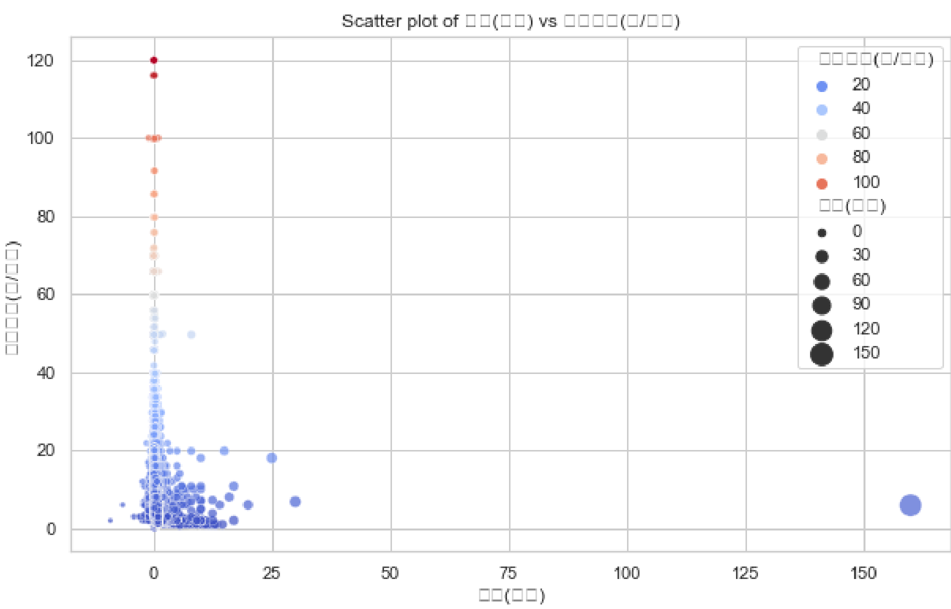

图 6-12: Spearman 秩相关性

初步结果显示各蔬菜品类的销售总量与成本加成定价的关系是负相关的关系。

接着使用了 Kendall's $\tau - b$ 相关性分析去对验证各蔬菜品类的销售总量与成本加成定价的关系。

6.3 Kendall's $\tau - b$ 相关性分析

Kendall's $\tau - b$ 相关性分析是一种非参数统计方法，用于度量两组排序数据之间的关联或一致性。与 Spearman 等级相关系数类似，Kendall 的 $\tau - b$ 考虑了数据中的等级。但与 Spearman 方法不同，Kendall 的 $\tau - b$ 是基于成对数据的一致性和不一致性来计算的。下面是 Kendall's $\tau - b$ 相关性分析的步骤：

1. 确定一致和不一致的对

选取两组品类数据 X 和 Y，对于任何两个观察值对 (x_i, y_i) 和 (x_j, y_j)：

• 如果 $x_i > x_j$ 并且 $y_i > y_j$ 或 $x_i < x_j$ 并且 $y_i < y_j$，那么这两对品类数据被认为是“一致的”。

• 如果 $x_i > x_j$ 并且 $y_i < y_j$ 或 $x_i < x_j$ 并且 $y_i > y_j$，那么这两对品类数据被认为是“不一致的”。

2. 计算 Kendall 的 τ

令 C 是一致对的数量，D 是不一致对的数量。Kendall's τ 的公式为：

$$\tau = \frac{C - D}{n(n-1)/2} \tag{1}$$

3. 考虑平级

在实际数据中，可能会存在相同或普及的观测值。为了考虑这些平级，Kendall 的 $\tau - b$ 考虑了它们。当平级存在时，τ 的公式进行了修正，从而得到 $\tau - b$。

4. 解释 $\tau - b$

$$\tau - b = \begin{cases} 1, & \text{完全正相关} \\ 0, & \text{没有相关性} \\ -1, & \text{完全负相关} \end{cases}$$

5. 显著性检验

如果需要检验 $\tau - b$ 是否显著不同于 0，可以使用相关的统计检验，通常涉及到 z 得分的计算。

6.3.1 结果展示与分析

由 SPSSPRO 得到结果

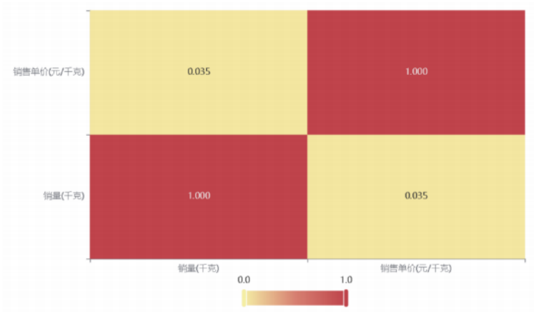

图 6-13: 花菜类

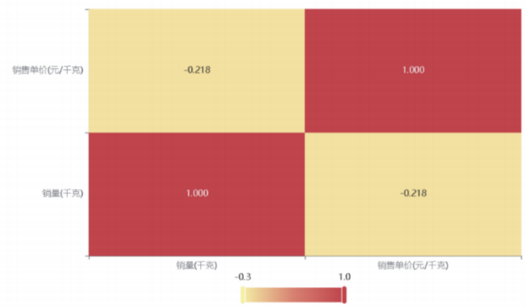

图 6-14: 花叶类

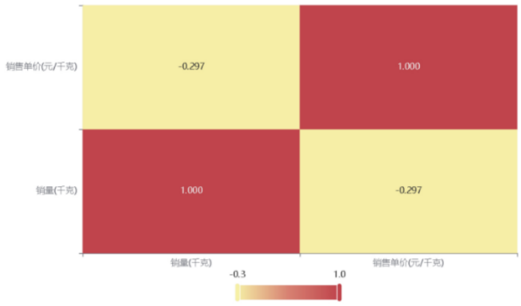

图 6-15: 茄类

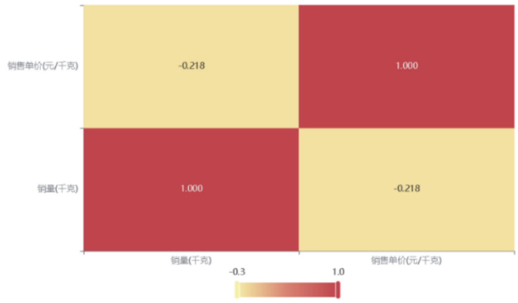

图 6-16: 辣椒类

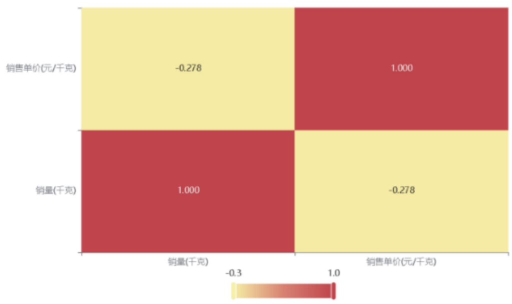

图 6-17: 食用菌

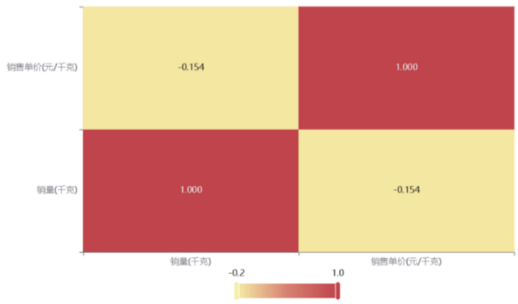

图 6-18: 水生根茎类

对于花菜类，它的销售总量与成本加成定价的关系是正相关，即成本加成的提高，会在一定范围里会使销售总量增加，但增量是较少的。对于花叶类，它的销售总量与成本加成定价的关系是负相关，即成本加成的提高，会在一定范围里会使销售总量降低。对于辣椒类，它的销售总量与成本加成定价的关系是负相关，即成本加成的提高，会在一定范围里会使销售总量降低。对于水生根茎类，它的销售总量与成本加成定价的关系是负相关，即成本加成的提高，会在一定范围里会使销售总量降低。对于食用菌，它的销售总量与成本加成定价的关系是负相关，即成本加成的提高，会在一定范围里会使销售总量降低。对于茄类，它的销售总量与成本加成定价的关系是负相关，即成本加成的提高，会在一定范围里会使销售总量降低。

综上所述，本文可以做出验证判断分析，六大品类的销售总量与成本加成定价的关系大致是负相关的。

6.4 基于时间序列预测模型解决收益最大化问题

题目要求我们求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。本文首先对未来一周的日补货总量进行预测。接着，进行定价策略的讨论。最后，通过动态规划进行对最大收益的求解。

6.4.1 时间序列分析模型预测日补货总量

根据本文前面的数据预处理，已经得到了六大品类 2023 年 6 月 1 日至 2023 年 6 月 30 日的数据。本文将继续使用这些数据，也就是 30 天的数据去预测未来一周的日补货总量。由于商家须在不确切知道具体单品和进货价格的情况下，做出当日各蔬菜品类的补货决策，所以本文考虑的是预测算法。至于为什么只采用 30 天的数据，一是蔬菜的供应品种在 4 月到 10 月是比较丰富的。六月和七月刚好在这个供应丰富的周期内，蔬菜品种的改变应该不大。二是我们只需要预测七天的数据，所以只需要选取短期的数据即可，能够提高预测的准确性。

6.4.2 模型的建立

指数平滑法是一种特殊的加权移动平均法。它是在移动平均法基础上发展起来的一种时间序列分析预测法，是通过计算指数平滑值，配合一定的时间序列预测模型对现象的未来进行预测。其原理是任一期的指数平滑值都是本期实际观察值与前一期指数平滑值的加权平均。根据平滑的次数不同，可分为一次指数平滑法和二次指数平滑法，或者更高次。当时间序列的变动出现直线趋势时，用一次指数平滑法进行预测，仍存在明显的滞后偏差。因此，也必须对其加以修正，再做二次指数平滑，利用滞后偏差的规律建立直线趋势模型。故本文使用的是二次指数平滑法，计算及推导过程如下：

$$\begin{cases} S_t^{(1)} = \alpha y_t + (1-\alpha) S_{t-1}^{(1)}, \\ S_t^{(2)} = \alpha S_t^{(1)} + (1-\alpha) S_{t-1}^{(2)} \end{cases} \tag{2}$$

式中 $S_t^{(1)}$ 为一次指数的平滑值;$S_t^{(2)}$ 为二次指数的平滑值. 当时间序列 $\{y_t\}$ 从某时期开始具有直线趋势时, 可用直线趋势模型

$$\hat{y}_{t+m} = a_t + b_t m, \quad m = 1, 2, \cdots \tag{3}$$

$$\begin{cases} a_t = 2S_t^{(1)} - S_t^{(2)} \\ b_t = \frac{\alpha}{1-\alpha}\left(S_t^{(1)} - S_t^{(2)}\right) \end{cases} \tag{4}$$

令 m=1, 可得

$$\hat{y}_{t+1}=2S_t^{(1)}-S_t^{(2)}+\frac{\alpha}{1-\alpha}\left(S_t^{(1)}-S_t^{(2)}\right) \tag{5}$$

如果数据不是平稳的，需要进行差分来获得平稳性。

A. 差分运算

差分运算是一种非常方便且有效的确定性信息提取方法。

$$\nabla^d X_t=(1-B)^d X_t=\sum_{i=0}^{d}(-1)^i C_d^i X_{t-i} \tag{6}$$

B. ARIMA 模型

对差分运算后得到的平稳序列可用 ARIMA 模型进行拟合。

$$\left\{\begin{array}{l}\phi(B)\nabla^d X_t=\theta(B)\varepsilon_t \\ E\left(\varepsilon_t\right)=0, \quad \operatorname{Var}\left(\varepsilon_t\right)=\sigma_{\varepsilon}^2, \quad E\left(\varepsilon_t\varepsilon_s\right)=0, \quad s\neq t \\ E\left(X_s\varepsilon_t\right)=0, \quad \forall s<t\end{array}\right. \tag{7}$$

6.4.3 结果展示与分析

1. 对于花叶类自相关与偏自相关

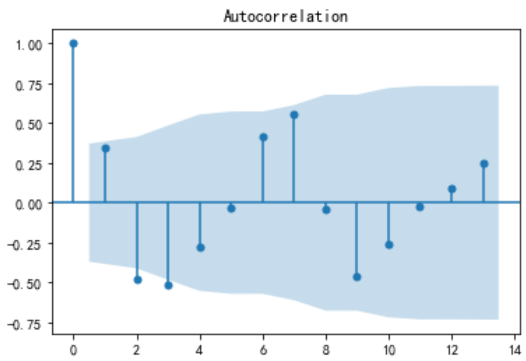

图 6-19: 花叶类自相关

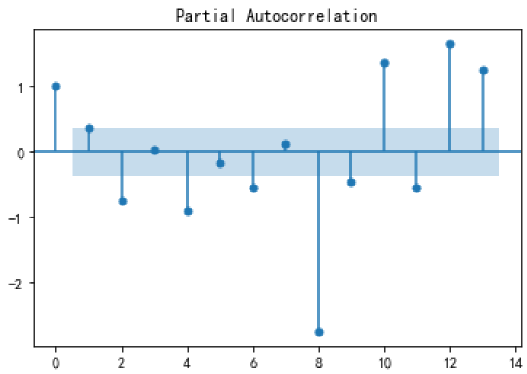

图 6-20: 花叶类偏自相关

注：两个图都是拖尾，其余自相关与偏自相关图片将放于附录。

2. 预测图

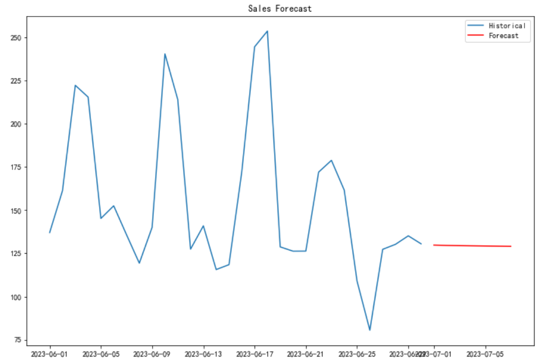

图 6-21：花叶类

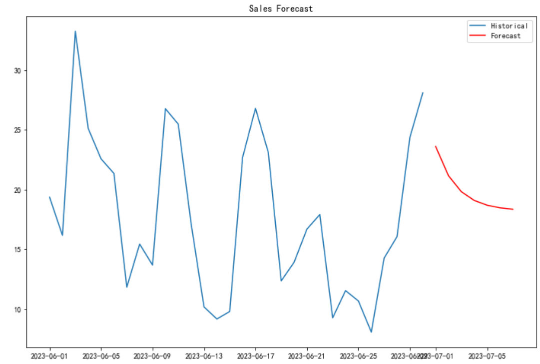

图 6-22：花菜类

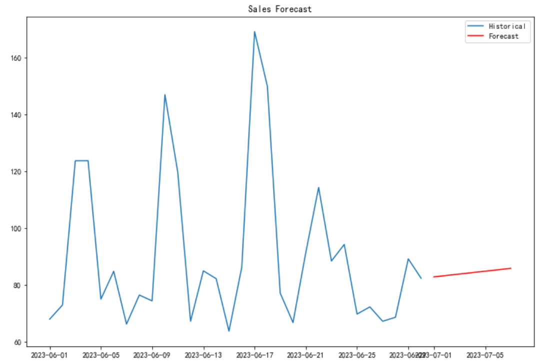

图 6-23：辣椒类

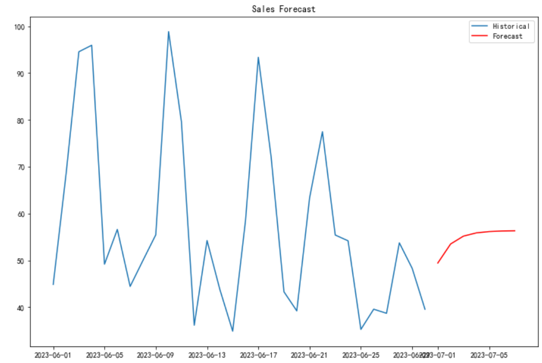

图 6-24：食用菌

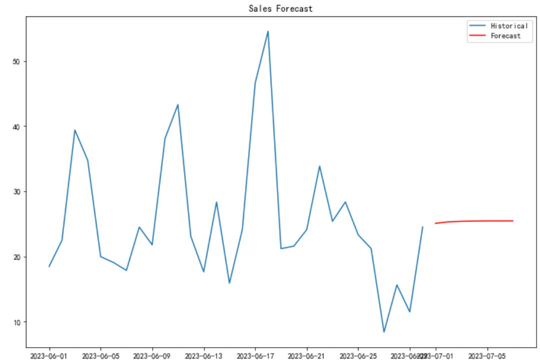

图 6-25：茄类

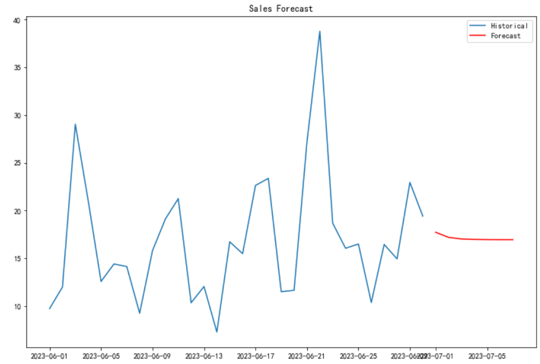

图 6-26：水生根茎类

3. 预测结果

表 6-3: 销量总览表

年份	预测日期	花叶类	辣椒类	食用菌	茄类	花菜类	水生根茎类
2023	07-01	129.74	82.77	49.44	25.09	23.60	17.73
	07-02	129.55	83.27	53.50	25.32	21.16	17.20
	07-03	129.43	83.77	55.18	25.41	19.82	17.03
	07-04	129.32	84.27	55.87	25.44	19.09	16.98
	07-05	129.21	84.77	56.15	25.46	18.69	16.96
	07-06	129.11	85.27	56.27	25.46	18.48	16.96
	07-07	129.00	85.77	56.32	25.47	18.36	16.95
2022	07-01	132.63	39.53	36.12	39.53	36.22	-
	07-02	133.57	42.51	35.66	42.51	32.46	-
	07-03	133.24	42.95	35.68	42.95	31.20	-
	07-04	133.36	43.02	35.68	43.02	30.79	-
	07-05	133.32	43.03	35.68	43.03	30.65	-
	07-06	133.33	43.03	35.68	43.03	30.60	-
	07-07	133.33	43.03	35.68	43.03	30.59	-
2021	07-01	136.59	-	-	-	28.30	9.97
	07-02	151.73	-	-	-	27.40	10.23
	07-03	153.20	-	-	-	27.04	10.41
	07-04	153.35	-	-	-	26.54	10.55
	07-05	153.36	-	-	-	26.08	10.65
	07-06	153.36	-	-	-	25.61	10.73
	07-07	153.36	-	-	-	25.14	10.78

注：2023 年 6 月数据预测的 7 月 1-7 日销量才是题目所求的补货量。

4. 预测分析

本文根据 2023 年 6 月份的销售量使用时间序列分析预测未来七天的日补货总量。在这里，本文作了一个假设，就是根据历史的销售量去假定未来的需求量，也就是补货量。从题目中可知，蔬菜类商品保质期较短，且品相随时间递增而变差，大部分品种无法隔日销售，所以不需要考虑库存问题。从整体来看，花叶类的日补货总量相对最大，水生根茎类的日补货总量相对最小。从补货量变化来看，第一，花叶类、茄类和水生根茎类的变化比较小，可以看作日补货总量维持稳定。第二，辣椒类和食用菌的日补货总量稳中有增，逐日递增，但增速不大。第三，花菜类的日补货总量稳中有降，逐

日递减，但负增速不大。

为了验证周期性与季节性，本文将采用 2021 年 6 月份和 2022 年 6 月份的数据去预测相应 7 月 1 日至 7 月 7 日的数据，然后与 2023 年预测的进行对比分析。最终发现，这三年的 7 月 1 日至 7 月 7 日的货量或者增速以及负增速相差不大，具有一定周期性。

6.5 定价策略

成本加成定价（cost-plus pricing）是一种确定产品或服务售价的简单方法，其基本思想是在产品或服务的成本基础上加上一个固定的利润率或加成金额来确定最终的售价。实际应用中分为：完全成本加成定价法、变动成本加成定价法、标准成本加成定价法。

在本文中，我们统计分析附件 2 的销量明细数据得知，该商超的定价策略是在批发价的基础上加上一定利润率来确定最终的售价，其中各品类的利润率各有不同且随时间变化利润率也有一定波动。因此，为了方便建模，本文我们将采用变动成本加成定价策略，其中加成率固定为三年历史销售数据统计得到的品类平均利润率。本文设定如下定价：

$$S_i = W_i(1 + m_i) \tag{8}$$

其中，i= 花菜类，花叶类，辣椒类，食用菌等，S_i 为品类的售价，W_i 为批发价，m_i 为品类的平均利润率，即加成率。

题中说明损坏和品相变差的商品将打折销售，我们根据历史销售数据可知，折扣率一般在五折上下波动，因此本文我们设定折扣系数 =0.5，则打折商品的定价公式为：

$$d_j = df_j \times w_j \times (1 + m_i) \tag{9}$$

其中，i= 花菜类，花叶类，辣椒类，食用菌等，j 表示单品，d_i 为折扣价，df_j 为折扣系数，W_j 为批发价，m_i 为品类的平均利润率，即加成率。

6.6 单目标规划模型求解最大收益

目标模型是数学建模中一种常用的模型，本身也涉及有最优采购的运筹求解。本文假设一些设定，以便简化复杂化的情况，但又能够完整的目标约束条件。

在前面，本文已经探究完了各蔬菜品类的销售总量与成本加成定价的关系，结论是负相关的关系，即成本加成定价的增加，在一定范围内会使销售总量减少。未来七天的日补货总量和定价策略已经确定，基于这些条件，接下来本文将采用单目标规划模型去求解商超的最大收益。

6.6.1 目标函数

本文在考虑收益最大时，不追求能够求出未来七天每天每种品类的最大收益，这样会导致有较大的误差且预测效果也不好，所以本文考虑追求期望收益最大。之所以是这样考虑，是因为在实际情况中，并不一定是每日的收益最大就能得到周收益最大。另外一种情况是期间可能产生一定的波动，但也能够达到期望最大收益。故本文采用后者探究方式去设立目标函数。

$$\text{期望最大收益} = \max \sum_{i=1}^{6} \sum_{j=1}^{251} \frac{[r_j \cdot s_{jt} \cdot (1 - l_j) + r_j \cdot d_i \cdot l_j] - r_j \cdot w_j}{7} \tag{10}$$

6.6.2 约束函数

$$d_i = df_j \cdot w_j \cdot (1 + m_{jt}) \tag{11}$$

$$m_{jt} = (t = 0, \text{正价率}; t = 1, \text{负价率}) \tag{12}$$

6.6.3 需求效益函数

需求效益函数也是约束条件之一。在这里作为额外假设引入。根据的是六大品类的销售总量与成本加成定价的关系大致是负相关的。

$$uf = a - b \times r_j \tag{13}$$

6.6.4 参数

表 6-4: 参数一览

批发价	whole price	W_j
销售价	sales price	S_{jt}
损耗率	loss rate	l_j
补货量	replenishment volum	r_j
销售空间	sale sapce	SS_j
加成率	markup rate	m_{jt}
效用函数	utility function	$uf = y = a - bx$
折扣价	discounted price	d_i
折扣系数	discount factor	df_j
品类	\	$i = 1, 2, 3, 4, 5, 6$
单品	\	$j = 1, 2, 3, \ldots, 251$
是否打折	\	t=(0,1)

6.6.5 整合

$$
期望最大收益 = \max \sum_{i=1}^{6} \sum_{j=1}^{251} \frac{[r_j \cdot s_{jt} \cdot (1 - l_j) + r_j \cdot d_i \cdot l_j] - r_j \cdot w_j}{7} \tag{14}
$$

$$
s,t \begin{cases} d_i = df_j \cdot w_j \cdot (1 + m_{jt}) \\ m_{jt} = (t = 0, 正价率; t = 1, 负价率) \\ uf = a - b \times r_j \end{cases} \tag{15}
$$

6.7 模型的求解结果

求解单目标规划问题得一周期望最大收益 (即 2023 年 7 月 1-7 日) 的近似值为 8113 元，即平均每天收益为 1159 元。

6.8 灵敏度分析

灵敏度分析是一种在定量研究中广泛使用的工具，主要用于评估模型、函数或系统输出对一个或多个输入参数的变化的响应。

在这里，本文为了减少复杂性，所以在此间选择进行单目标规划的灵敏度分析，即以销售单价 (元/千克) 在扰动范围 [-0.1,0.1] 之间探究它与销量 (千克) 之间的灵敏度关系。使用 Python 编程求解如下：

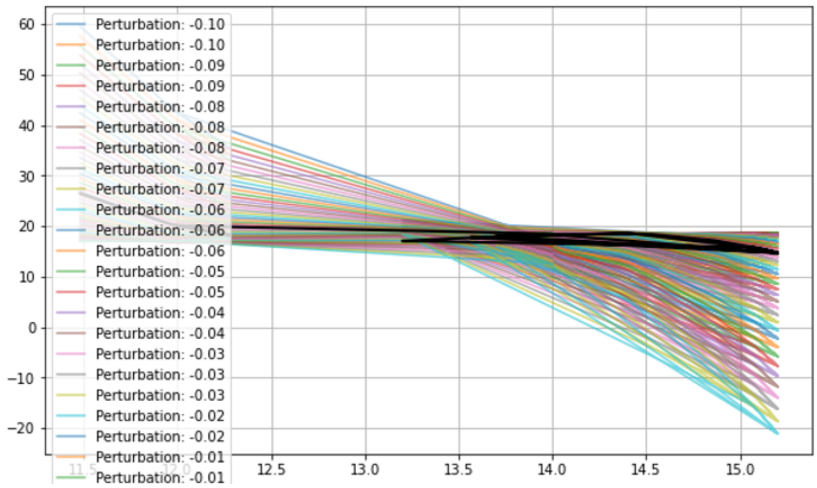

图 6-27: 灵敏度分析

本文的结论是两者之间的关系是存在较灵敏的，从而也再次验证了六大品类的销售总量与成本加成定价的关系大致是负相关的。

七、 问题三的模型建立与求解

7.1 数据预处理

根据题目要求，由于蔬菜类商品的销售空间有限，需要进一步制定单品的补货计划，且按照 2023 年 6 月 24-30 日的可售品种来规划 7 月 1 日的单品补货量和定价策略，所以本文截取 2023 年 6 月 24 到 2023 年 6 月 30 日的数据。

7.2 量化分析

由于 7 月 1 日的单品补货量和定价策略是参考 2023 年 6 月 24 到 2023 年 6 月 30 日的数据，所以有必要对这些数据进行量化分析。

7.2.1 对品类的量化分析

表 7-5: 7 月 1-7 日各品类的销售总量

品类	花叶类	花菜类	辣椒类	食用菌	茄类	水生根茎类
总销售量 (kg)	873.928	113.107	543.083	309.369	133.015	116.702

表中是这一周内的各品类的销售总量，从中可以看出，六个品类都有销售。其中，大致可以看出，花叶类的销售量是最大的，水生根茎类和茄类的销售量相对较少。

7.2.2 对单品的量化分析

表 7-6: 7 月 1-7 日排名前 49 的单品的销售总量

排名	单品名称	销量 (kg)	排名	单品名称	销量 (kg)
1	云南生菜	226	26	上海青	27.029
2	小米椒 (份)	150	27	枝江青梗散花	25.203
3	云南油麦菜	149	28	高瓜 (1)	21.031
4	金针菇 (盒)	113	29	青茄子 (1)	18.727
5	芜湖青椒	99.634	30	虫草花 (份)	16
6	竹叶菜	93.077	31	青红杭椒组合装 (份)	15
7	西兰花	87.904	32	红椒 (2)	14.25
8	小皱皮 (份)	79	33	菱角	12.224
9	螺丝椒 (份)	79	34	云南生菜	11.683
10	紫茄子 (2)	76.32	35	菜心	8.877

11	娃娃菜	73	36	蟹味菇与白玉菇双拼 (盒)	8
12	双孢菇 (盒)	70	37	圆茄子 (2)	7.443
13	苋菜	62.461	38	七彩椒 (2)	7.302
14	海鲜菇 (包)	62	39	菠菜	6.82
15	姜蒜小米椒组合装 (小份)	49	40	外地茼蒿	6.538
16	菠菜 (份)	49	41	高瓜 (2)	5.996
17	螺丝椒	47.897	42	云南油麦菜	5.098
18	奶白菜	44.959	43	红莲藕带	4.596
19	净藕 (1)	42.184	44	鲜木耳 (份)	4
20	木耳菜	41.534	45	白玉菇 (袋)	4
21	小青菜 (1)	34.306	46	木耳菜 (份)	3
22	西峡花菇 (1)	32.369	47	野生粉藕	2.439
23	红薯尖	31.546	48	青线椒 (份)	2
24	长线茄	29.501	49	紫茄子 (1)	1.024
25	洪湖藕带	28.232	50	\	\

根据题目的要求，各单品订购量满足最小陈列量 2.5 千克的要求，即单品的每天的订购量要大于等于 2.5 千克。因为以上排名是过去一周的销量排名，所以本文初步挑选出销量大于 17.5 千克。单品如果它小于 17.5 千克，说明每日销售不大于 2.5 千克，说明这种单品销售情况不太理想或者目前需求不大，不需要订购大于 2.5 千克，进而本文就不再 7 月 1 日考虑它们的订购了。

至此，本文初步选取了销售量排名前 29 的单品，排除了 30 名及以后的单品来提高商超的收益。接着，本文继续建立回归分析模型来预测 7 月 1 日的订购量。

7.3 回归分析模型预测订购量

回归分析是确定两种或两种以上变量间相互依赖的定量关系的一种统计分析方法。使用 Python 编程拟合回归方程：

表 7-7: 7 月 1 日单品的销售总量

单品名称	拟合方程	R^2	销量 (kg)
云南生菜 (份)	$y = 3.18x^2 - 26.89x + 76.29$	0.571	64.69
小米椒 (份)	$y = 0.30x^2 - 0.99x + 19.43$	0.348	30.71
云南生菜	$y = -0.35x^2 + 2.82x - 1.95$	0.641	-1.79
金针菇 (盒)	$y = 0.02x^2 + 0.81x + 12.43$	0.505	20.19
芜湖青椒 (1)	$y = 0.43x^2 - 3.39x + 19.21$	0.282	19.61
竹叶菜	$y = 0.18x^2 - 2.07x + 17.92$	0.599	12.88
西兰花	$y = 0.12x^2 + 0.37x + 8.71$	0.789	19.35
小皱皮 (份)	$y = 0.38x^2 - 4.05x + 19.86$	0.374	11.78
螺丝椒 (份)	$y = 0.64x^2 - 5.14x + 19.00$	0.253	18.84
紫茄子 (2)	$y = 0.67x^2 - 6.40x + 23.19$	0.386	14.87
娃娃菜	$y = -0.73x^2 + 6.35x - 0.43$	0.252	3.65
双孢菇 (盒)	$y = -0.08x^2 + 0.35x + 10.29$	0.524	7.97
苋菜	$y = 0.32x^2 - 3.57x + 16.85$	0.565	8.77
海鲜菇 (包)	$y = 0.25x^2 - 2.68x + 14.57$	0.256	9.13
姜蒜小米椒组合装 (小份)	$y = 0.11x^2 - 1.18x + 9.57$	0.529	7.17
菠菜 (份)	$y = -0.71x^2 + 6.84x - 3.72$	0.821	5.56
螺丝椒	$y = 0.40x^2 - 3.09x + 11.28$	0.721	12.16
奶白菜	$y = 0.03x^2 + 0.01x + 6.77$	0.259	8.77
净藕 (1)	$y = 0.24x^2 - 1.79x + 8.47$	0.459	9.51
木耳菜	$y = -0.46x^2 + 3.94x - 0.56$	0.436	1.52
小青菜 (1)	$y = 0.21x^2 - 1.67x + 7.46$	0.589	7.54
西峡花菇 (1)	$y = 0.32x^2 - 2.52x + 8.27$	0.954	8.59
红薯尖	$y = 0.04x^2 + 0.01x + 3.64$	0.598	6.28
长线茄	$y = 0.60x^2 - 4.41x + 9.92$	0.746	13.04
洪湖藕带	$y = -0.06x^2 + 0.68x + 2.48$	0.297	4.08
上海青	$y = 0.39x^2 - 2.52x + 6.17$	0.710	10.97
枝江青梗散花	$y = 0.60x^2 - 3.18x + 4.66$	0.911	17.62
高瓜 (1)	$y = 0.06x^2 - 0.31x + 3.03$	0.334	4.39
青茄子 (1)	$y = 0.29x^2 - 3.07x + 9.53$	0.869	3.53

从拟合的回归方程来看，拟合的效果良好，接下来本文还将继续进行添加扰动范围来修正回归方程预测所带来的误差。经过查阅相关文献，回归方程的误差扰动范围取值在 [-0.12,0.12] 较好。

7.4 定价策略

本文基于问题二的定价策略,对上一周的销售数据进行分析，找出销售情况良好的商品。对于上一周销售情况良好的商品，提高其售价。

调整后售价 = 基础售价 +β× 基础售价

其中，β 是基于销售表现的加成率。

7.5 双目标规划模型

题目要求我们在尽量满足市场对各品类蔬菜商品需求的前提下，使得商超的收益最大。本问与问题二的单目标规划情况大致相同，基于此，在此引入双目标规划模型。即现在有两个未知量，一个是期望收益，另一个是补货量。注意，本文在前面使用回归方程求出来的只是一个大致的结果，并不代表着最终的补货量。为此，本文将引入扰动范围。故现在补货量也是一个未知量。根据题目要求，继续添加一些约束条件。

第一个是放置空间，由于商超的摆货空间并不是无限大的，所以有一定的限制空间。

$$\text{假定} ps \leq 10000 \tag{16}$$

第二个是资金空间，商超的订货资金也不是无限多的，所以也有一定的限制空间。

$$\text{假定} fs \leq 5000 \tag{17}$$

第三个是消费者选择因子，存在一定概率消费者当天可能会选择别家商超去购买随菜类商品，所以该因子 mf 是一个随机数。

第四个是扰动范围。

$$R_j = y_i \times \alpha \tag{18}$$

，其中 $\alpha = [-0.12, 0.12]$。

第五个是单品的限制，经过本文前面的选择，目前只剩下 29 种商品。故 $27 \leq s_i \leq 29$。

7.5.1 整合

$$\text{期望最大收益} = \max \sum_{i=1}^{6} \sum_{j=1}^{251} \frac{[r_j \cdot s_{jt} \cdot (1 - l_j) + r_j \cdot d_i \cdot l_j] - r_j \cdot w_j}{7} \tag{19}$$

$$
s,t\begin{cases}
d_i = df_j \cdot w_j \cdot (1+m_{jt}) \\
m_{jt} = (t=0,\text{正价率};t=1,\text{负价率}) \\
uf = a - b\cdot r_j \\
ps{\leq}10000 \\
fs{\leq}5000 \\
mf \\
R_j = y_i \cdot \alpha
\end{cases} \tag{20}
$$

7.5.2 模型的求解结果

求解双目标规划模型得 7 月 1 日的期望最大收益的近似值为 1359 元。

八、 问题四的思考与建议

为了更好地制定蔬菜商品的补货和定价决策，本文建议商超可以采集以下相关数据：

1. 消费者反馈数据

通过收集消费者的反馈数据，可以收集到一些消费者需要的蔬菜种类，但是商超没有进行售卖的品种。使得商超通过提高商品种类，从而提高商品的销售量。另外，可以收集到消费者对于商品的价格的态度，对商品的价格进行调整，使得商品销售量提高的同时商品的价格又不低，有益于商超的收益率。

2. 库存数据

收集每一天的库存数量可以使得商超了解每一种商蔬菜商品的最近的销售情况，也同时可以了解蔬菜商品的剩余数量，可以让商超知道什么时候应该补哪一种蔬菜商品以及对应的补货数量，这样可以减少蔬菜商品的损耗率，也可以使得商超不需要因为蔬菜品相而打折销售，可以减少蔬菜商品的成本，也就可以提高商超的利润率，从而提高商超的收益。

3. 货品损耗数据

除了损耗率，还要记录货品损耗的具体原因，如是否是因为存储环境导致的，则需要改进存储方法，减少损耗。

4. 天气预报数据

各类蔬菜的销售很可能受天气的影响。例如，在广州寒冷天气可能促使人们购买更多的用于炖汤的蔬菜。同时，在重庆寒冷的天气则可能使得人们购买辣椒进行打火锅，这可以帮助商超对各类蔬菜商品的销售量进行预测，使得供应量远远大于需求量的情况或者需求量远远大于供应量，这

使得商超的收益不能得到提高。同时也可以为商超各类商品的补货和商品的定价提供参考。

5. 附近商超的信息

收集附近商超的相关数据。比如，商超的促销活动，对于商家的促销活动一般时间都是比较稳定的。当对手商家进行促销活动，商超选择同时进行促销活动，那么在商品的定价上则需要考虑对手商超的定价。不选择同时进行促销活动时，当同一种商品在其他商超更优惠时，那么商超的销售量一定会下降，那么也就为商超的补货量提供了参考了。

以上这些方面的数据可以帮助商超更准确地预测蔬菜的需求、降低损耗和更有针对性地定价，从而提高整体的盈利能力。

九、 模型的评估

9.1 模型的优点

1. 数据处理将 3 年每一天的数据转换为每个月的平均数据，远远减少了数据量的同时数据的代表性没有受到太大的影响。
2. 本文的各种模型全面，数据可视化绘，图简洁美观文字与图形相表述更易理解。
3. 本文的代码简洁且有效。
4. 结合实际情况进行分析，使得结果更具有说服力。

9.2 模型的缺点

1. 模型假设比较理想化，在实际生活中较难实现。
2. 同一单品不同供应商我们假定为不同的商品，减少数据处理量的同时数据的准确度也有所下降。

参考文献

[1] 胡彬. 煤炭企业以竞争为导向的定价策略研究 [J]. 内蒙古煤炭经济，2013(09).

[2] 申忠梅. 商业银行产品定价策略探讨 [J]. 现代金融，2003(12).

[3] 周争，石陆华，汪洁滢，陆秉炜，杨晓秋，仓艺倩，丁文彬，胡丹，王青，王争，董菡F，顾乐怡，戴慧莉. 基于多因素时间序列的医院科研绩效预测模型构建与应用 [J]. 北大核心，2022,42(10).

[4] 王珊珊，丁浩，王占锋，带有非参随机效应的稳健函数型回归模型（英文）[J]. 中国科学技术大学学报，2022，52(04).

[5] 陆俊平，谢新乔，李湘伟，杨继周，王剑松，田育天，徐梓荷，朱云聪. 玉溪烟区土壤主要理化性状与烟叶品质的相关性分析 [J]. 黑龙江农业科学，2022(10).

[6] 陈家辉，蔡磊，余坤江. 甘蓝型油菜种质群体 6 个农艺性状变异及相关性分析 [J]. 山地农业生物学报，2022,41(05).

[7] 司守奎，孙玺菁著.Python 数学实验与建模, 科学出版社，2020 年 4 月第一版.

[8] 姜雅莉. 蔬菜价格波动及传导研究 [C]. 西北农林科技大学，2013 年第 12 期.

[9] 潘晓飞，解志恒. 王淑云考虑损失规避的生鲜品商超保鲜努力和定价的优化决策 [J]. 公路交通科技,2022,39(06).

[10] 李晓璐，周曙光. 我国生鲜商超零售业发展问题研究 [J]. 商业经济研究，2021(23).

附　录 A

问题二中品类自相关和偏自相关的结果展示

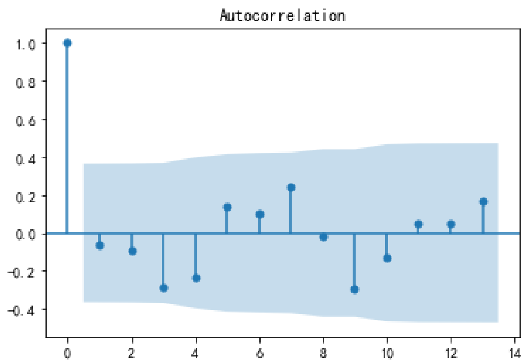

图 9-28: 花菜类自相关

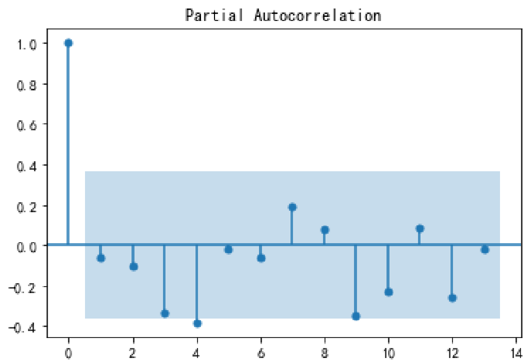

图 9-29: 花菜类偏自相关

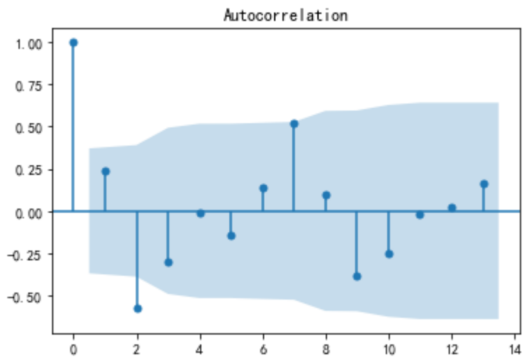

图 9-30: 辣椒类自相关

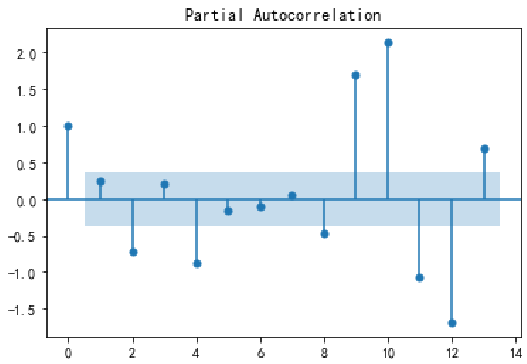

图 9-31: 辣椒类偏自相关

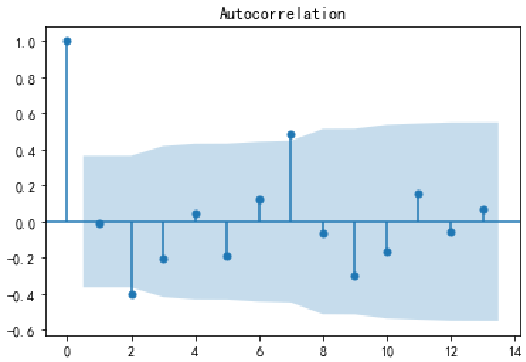

图 9-32: 食用菌自相关

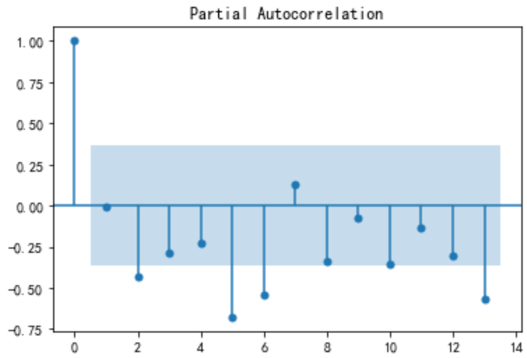

图 9-33: 食用菌偏自相关

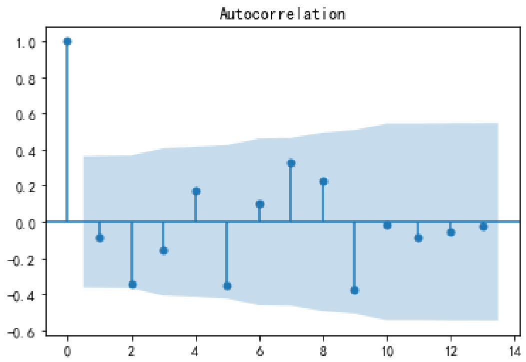

图 9-34: 茄类自相关

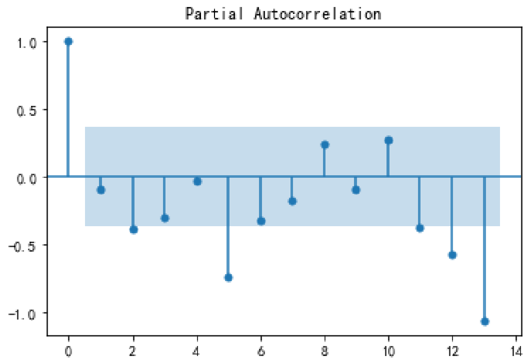

图 9-35: 茄类偏自相关

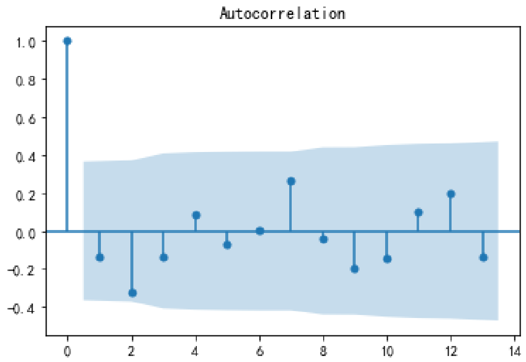

图 9-36: 水生根茎类自相关

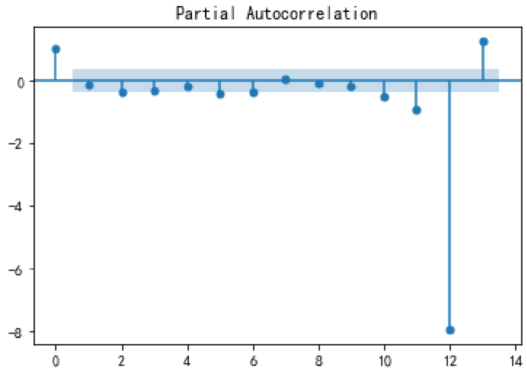

图 9-37: 水生根茎类偏自相关

附 录 B

代码一：预处理时间饼图

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

# 读取Excel文件
df = pd.read_excel('D:\ipy\附件2.xlsx', engine='openpyxl')

# 转换为时间格式并提取小时数据
df['Hour'] = pd.to_datetime(df['扫码销售时间']).dt.hour

# 对小时数据进行计数
hour_counts = df['Hour'].value_counts().sort_index()

# 绘制饼图
```

```python
fig, ax = plt.subplots(figsize=(10, 6))
ax.pie(hour_counts, labels=hour_counts.index, autopct='%1.1f%%', startangle=90)
ax.axis('equal')  # Equal aspect ratio ensures that pie is drawn as a circle.

plt.title('24 Hours Distribution of Scan Sales')
plt.show()
```

代码二：单品销量排名

```python
import pandas as pd

# 读取Excel文件
df = pd.read_excel('D:\ipy\Q1附件2增加名称.xlsx', engine='openpyxl')

# 根据单品名称和分类名称分组并求销量的和
grouped = df.groupby(['单品名称', '分类名称'])['销量(千克)'].sum().reset_index()

# 根据销量从高到低排序
sorted_df = grouped.sort_values(by='销量(千克)', ascending=False)

# 保存结果到新的Excel文件
sorted_df.to_excel('Q1单品销量排名.xlsx', index=False, engine='openpyxl')
```

代码三：问题一中 Python 对单品销量逐月量化

```python
import pandas as pd
import matplotlib.pyplot as plt
plt.rcParams['font.family']='SimHei'        # 设置使用黑体字体以正常显示中文
plt.rcParams['axes.unicode_minus']=False    # 正常显示负号

# 读取Excel文件
df = pd.read_excel('D:\ipy\Q1附件2增加名称.xlsx', engine='openpyxl')

# 转换销售日期为日期格式
df['销售日期'] = pd.to_datetime(df['销售日期'])
```

```python

# 筛选日期
date_filter = (df['销售日期'] >= '2022-07-01') & (df['销售日期'] <= '
    2023-06-30')
df = df[date_filter]

# 单品名称筛选
product_names = ['芜湖青椒(1)', '西兰花', '净藕(1)', '大白菜', '金针
    菇(盒)', '紫茄子(2)']
df = df[df['单品名称'].isin(product_names)]

# 根据销售日期的月份和单品名称分组并求销量的和
df['YearMonth'] = df['销售日期'].dt.to_period('M')
grouped = df.groupby(['YearMonth', '单品名称'])['销量(千克)'].sum().
    unstack().fillna(0)

# 绘制条形图
ax = grouped.plot(kind='bar', figsize=(14, 7))
plt.title('Monthly Sales from 2022/7/1 to 2023/6/30')
plt.xlabel('Year-Month')
plt.ylabel('Total Sales (kg)')
plt.legend(title='Product Name', bbox_to_anchor=(1.05, 1), loc='upper
     left')

# 添加数据注释
for i, rect in enumerate(ax.patches):
    height = rect.get_height()
    ax.text(rect.get_x() + rect.get_width() / 2, height, '{:.2f}'.
        format(height),
            ha='center', va='bottom', rotation=90)

plt.tight_layout()
plt.show()
```

代码四：问题二时间序列分析

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from statsmodels.tsa.stattools import adfuller
```

```python
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
from statsmodels.tsa.arima.model import ARIMA

# 读取数据
data = pd.read_excel('D:\ipy\Q2\食用菌21.xlsx.xlsx', engine='openpyxl')
data['日期'] = pd.to_datetime(data['日期'])
data.set_index('日期', inplace=True)

def test_stationarity(timeseries):
    #Determing rolling statistics
    rolmean = timeseries.rolling(12).mean()
    rolstd = timeseries.rolling(12).std()

    #Plot rolling statistics:
    plt.figure(figsize=(12, 8))
    orig = plt.plot(timeseries, color='blue',label='Original')
    mean = plt.plot(rolmean, color='red', label='Rolling Mean')
    std = plt.plot(rolstd, color='black', label = 'Rolling Std')
    plt.legend(loc='best')
    plt.title('Rolling Mean & Standard Deviation')
    plt.show()

    #Perform Dickey-Fuller test:
    result = adfuller(timeseries, autolag='AIC')
    dfoutput = pd.Series(result[0:4], index=['Test Statistic','p-value','#Lags Used','Number of Observations Used'])
    for key,value in result[4].items():
        dfoutput['Critical Value (%s)'%key] = value
    print(dfoutput)

test_stationarity(data['销量(千克)'])

diff_n = 0
p_value = 1
while p_value > 0.05:
    diff_n += 1
    diff_data = data['销量(千克)'].diff(diff_n).dropna()
    p_value = adfuller(diff_data)[1]
```

```python

data_diff = data['销量(千克)'].diff(diff_n).dropna()

max_lags = min(15, len(data_diff) // 2 - 1)
plot_acf(data_diff, lags=max_lags)
plot_pacf(data_diff, lags=max_lags)
plt.show()

model = ARIMA(data['销量(千克)'], order=(1, diff_n, 1))
model_fit = model.fit()
forecast = model_fit.forecast(steps=7)

plt.figure(figsize=(12, 8))
plt.plot(data.index, data['销量(千克)'], label='Historical')
plt.plot(pd.date_range(data.index[-1], periods=8, closed='right'),
    forecast, color='red', label='Forecast')
plt.legend()
plt.title('Sales Forecast')
plt.show()

print("Forecast for the next 7 days:")
print(forecast)
```

代码五：问题三量化分析

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

# 读取Excel数据
input_filename = 'D:\ipy\Q3\六月24-30数据.xlsx'  # 修改为你的Excel文
    件路径
output_filename = 'Q3量化分析.xlsx'  # 输出文件的路径和名称
df = pd.read_excel(input_filename, engine='openpyxl')

# 1. 统计分类名称的销量并绘制饼图
category_sales = df.groupby('分类名称')['销量(千克)'].sum()

# 打印每个分类名称的销量
print(category_sales)

```

```
# 绘制饼图
plt.figure(figsize=(10, 7))
category_sales.plot.pie(autopct='%1.1f%%', startangle=90)
plt.title("Sales Distribution by Category")
plt.ylabel('')
plt.show()

# 2. 统计每个分类名称中的单品名称的全部销量并排序
product_sales = df.groupby(['分类名称', '单品名称'])['销量(千克)'].
    sum().reset_index()
sorted_product_sales = product_sales.sort_values(by='销量(千克)',
    ascending=False)

# 保存到新的Excel文件
sorted_product_sales.to_excel(output_filename, index=False, engine='
    openpyxl')

print(f"Sorted sales data saved to {output_filename}")
```

代码六：问题三拟合方程

```
import pandas as pd
import numpy as np

# 读取Excel数据
input_filename = 'D:\ipy\Q3\Q3量化日期转化数字.xlsx'  # 修改为你的
    Excel文件路径
df = pd.read_excel(input_filename, engine='openpyxl')

# 统计每个单品名称的每天的销量
product_daily_sales = df.groupby(['单品名称', '日期'])['销量(千克)'].
    sum().reset_index()

# 对于每个单品，进行二次回归拟合
regression_results = {}

for product in product_daily_sales['单品名称'].unique():
    subset = product_daily_sales[product_daily_sales['单品名称'] ==
        product]

```

```python
    x = subset['日期'].values
    y = subset['销量(千克)'].values

    if len(x) < 3:  # 二次拟合至少需要3个数据点
        print(f"Skipping product '{product}' due to insufficient data.")
        continue

    # 使用numpy进行二次拟合
    coefficients, residuals, _, _, _ = np.polyfit(x, y, 2, full=True)

    if len(residuals) == 0:
        print(f"Unable to fit regression for product '{product}'.")
        continue

    a, b, c = coefficients
    # 计算R^2
    total_var = ((y - y.mean())**2).sum()
    r_squared = 1 - residuals[0] / total_var

    regression_results[product] = (a, b, c, r_squared)

# 打印回归方程结果和R^2值
for product, (a, b, c, r2) in regression_results.items():
    print(f"For product '{product}': y = {a:.2f}x^2 + {b:.2f}x + {c:.2f}, R^2 = {r2:.3f}")
```