授权给:

**C** 题 蔬菜类商品的自动定价与补货决策

问题 **1** 蔬菜类商品不同品类或不同单品之间可能存在一定的关联关系，请分析蔬菜各

品类及单品销售量的分布规律及相互关系。

问题 **2** 考虑商超以品类为单位做补货计划，请分析各蔬菜品类的销售总量与成本加成 定价的关系，并给出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。

问题 **3** 因蔬菜类商品的销售空间有限，商超希望进一步制定单品的补货计划，要求可 售单品总数控制在 27-33 个，且各单品订购量满足最小陈列量 2.5 千克的要求。根据

2023

年 6 月 24-30 日的可售品种，给出 7 月 1 日的单品补货量和定价策略，在尽量满足

市场对各

品类蔬菜商品需求的前提下，使得商超收益最大。

问题 **4** 为了更好地制定蔬菜商品的补货和定价决策，商超还需要采集哪些相关数据，

这些数据对解决上述问题有何帮助，请给出你们的意见和理由。

整体分析：

需要运用统计分析方法处理历史数据,并建立线性规划数学模型,以收益最大化

为目标,综合考虑各种业务约束,求解最优的补货和定价策略。同时,需要思考额

外的数据对确定最优决策的作用。

对问题 1,使用统计分析方法,分析不同蔬菜品类和单品之间的销量分布规律和关

联关系。这可以帮助后续确定重点品类和单品。

对问题 2,建立优化模型, 以最大化商超收益为目标,综合考虑销量、成本、定价等

因素,给出一个星期内每个品类的最佳日补货量和定价策略。

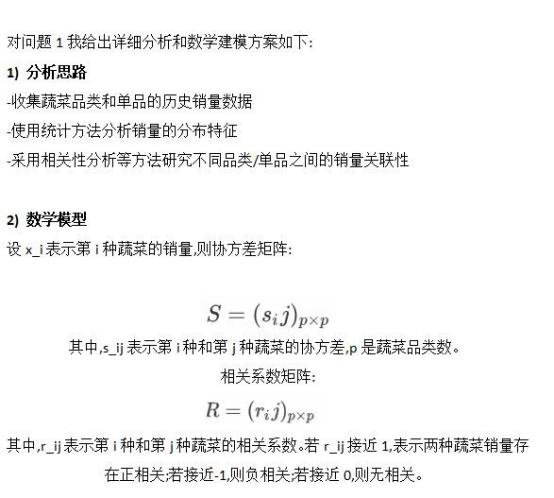
对问题 3,将添加限制条件,使售卖的单品总数和最小陈列量在给定范围内。在此

基础上给出单品层面的补货量和定价计划, 以最大化收益。

对问题 4,考虑需要补充获取的信息, 比如客户购买习惯、促销效果等数据,分析这

些数据如何帮助建立更准确的预测模型,从而制定更好的决策方案。

问题一详细思路分析**+**建模步骤**+**求解参考代码：



3) 求解代码(Python)```python

python import pandas as pdimport numpy as npimport prop lot as ppltimport seaborn as snsfrom scipy.stats import pearsonrimport matplotlib.pyplot as pltfrom matplotlib.ticker import FormatStrFormatter

import numpy as np# 导入销量数据

X = pd.read\_excel('附件 2.xlsx')# 计算协方差矩阵

S = np.cov(X)

# 计算相关系数矩阵

R = np.corrcoef(X)

# 分析相关系数大小判断相关性

from scipy.stats import pearsonr

pvals = heatmap\_data.corr (method=lambda x, y: pearsonr (x, y)[1]) -

np.eye(len (heatmap\_data.columns))

#转换 P 值为星号 def convert\_pvalue\_to\_asterisks(pvalue) :

if pvalue <= 0.001:

return "\*\*\*"

elif pvalue <= 0.01:

return "\*\*"

elif pvalue <= 0.05:

return "\*"

return ""

pval\_star = pvals.applymap (lambda x :convert\_pvalue\_to\_asterisks(x))

# 转换成 numpy 类型

corr\_star\_annot = pval\_star.to\_numpy ()

# a）seaborn 相关性矩阵热力图 p 值绘制示例一

import matplotlib.pyplot as pltfrom colormaps import parulafrom matplotlib.ticker import FormatStrFormatter

fig,ax = plt.subplots(fig size= (4,3.5),dpi=100,facecolor="w")

sns.heatmap (heatmap\_data.corr (),annot=corr\_star\_annot,fmt='',cmap=parula,vmin=-1, vmax=1,

annot\_kws= {"size":13,"fontweight":"bold"},linecolor="k",linewidths=.2

,

cbar\_kws= {"aspect":13},ax=ax)

ax.tick\_params(bottom=False,

labelbottom=True,labeltop=False,left=False,pad=1,labelsize=12)

ax.yaxis.set\_tick\_params(labelrotation=0)# 使用 matplotlib.colorbar.Colorbar object

cbar = ax.collections[0].colorbar #分享群：810589811

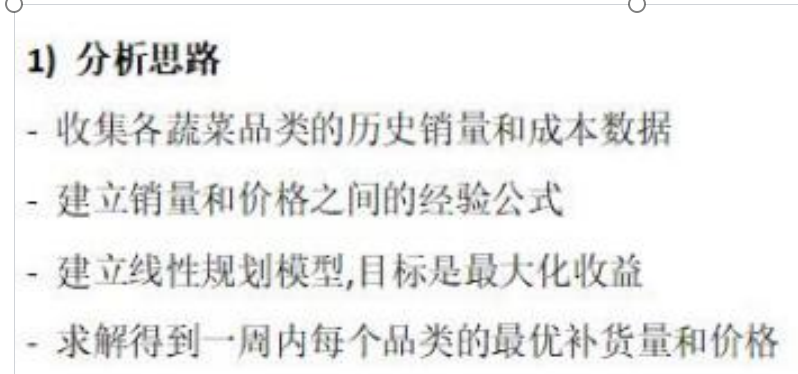
cbar.ax.tick\_params(direction="in",width=.5,labelsize=10)

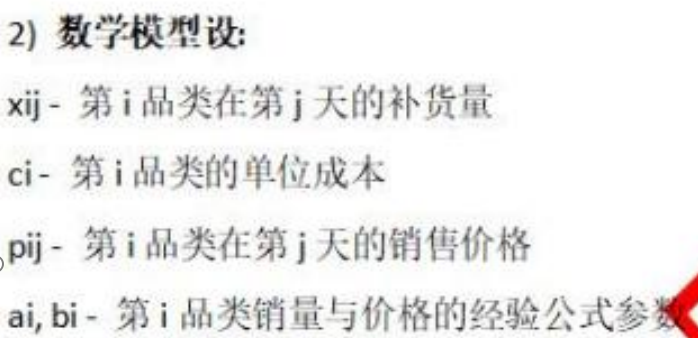
cbar.ax.yaxis.set\_major\_formatter (FormatStrFormatter ('%.2f'))

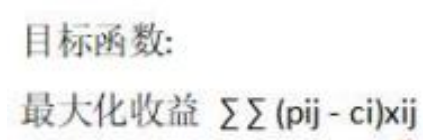
cbar.outline.set\_visible(True)

cbar.outline.set\_linewidth(.5)

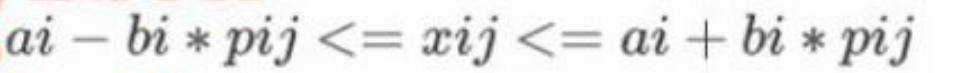
问题二详细思路分析**+**建模步骤**+**求解参考代码：







约束：



参考求解代码**(Python):**

python

from pulp import \*

# 构建模型

model = LpProblem()

# 定义决策变量

x = {}

p = {}

for i in brands:

for j in days:

x[i,j] = LpVariable()

p[i,j] = LpVariable()

# 添加目标函数和约束条件

model += lpSum((p[i,j] - c [i])\*x[i,j]) # 最大化收益

for i in brands:

for j in days:

model += a [i] - b[i]\*p[i,j] <= x[i,j] <= a [i] + b[i]\*p[i,j] # 销量与价格约束

python

# 在问题 2 代码基础上添加:

# 定义单品售卖变量 y

y = {}

for i in products:

for j in days:

y[i,j] = LpVariable(cat='Binary')

# 添加单品数量限制

model += lpSum(y[i,j] for i in products for j in days) >= 27

model += lpSum(y[i,j] for i in products for j in days) <= 33

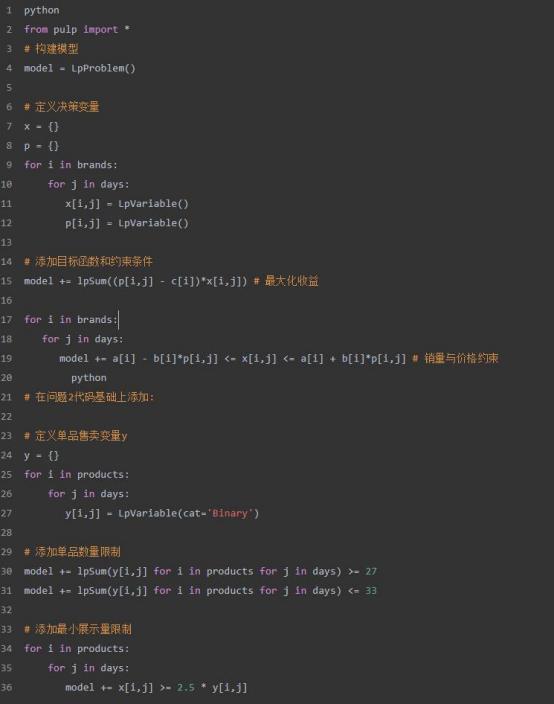
# 添加最小展示量限制

for i in products:

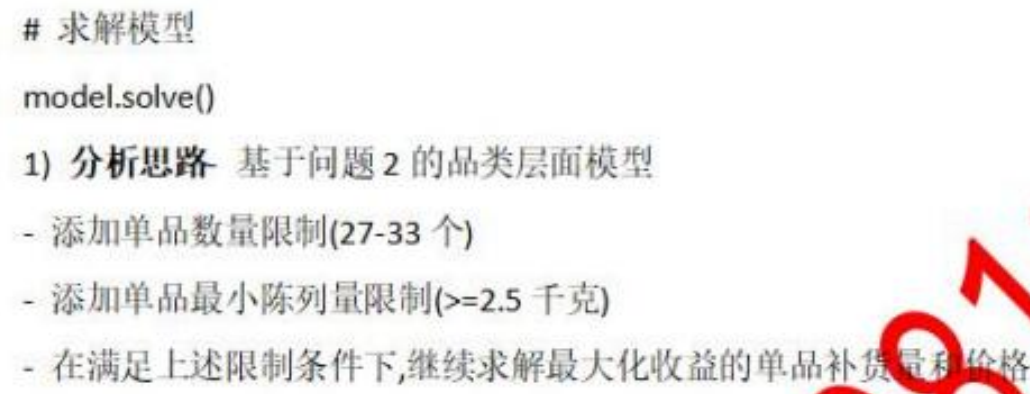
for j in days:

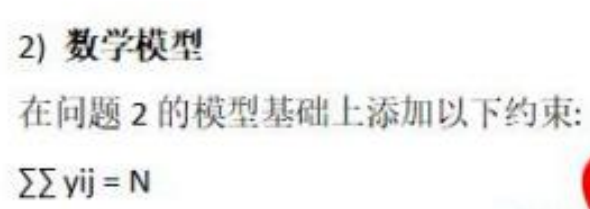
model += x[i,j] >= 2.5 \*

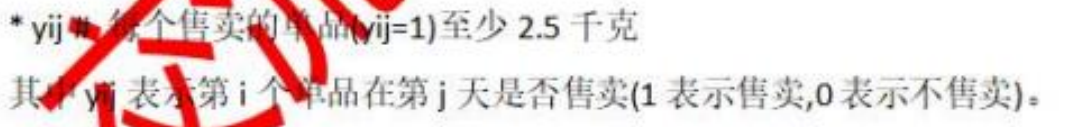
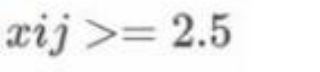
y[i,j]



问题三详细思路分析**+**建模步骤**+**求解参考代码：







参考求解代码**(Python):**

python

# 在问题 2 代码基础上添加:

# 定义单品售卖变量 y

y = {}

for i in products:

for j in days:

y[i,j] = LpVariable(cat='Binary')

# 添加单品数量限制

model += lpSum(y[i,j] for i in products for j in days) >= 27

model += lpSum(y[i,j] for i in products for j in days) <= 33

# 添加最小展示量限制

for i in products:

for j in days:

model += x[i,j] >= 2.5 \* y[i,j]

