

# 基于多重规划的生产企业原材料的订购与运输方案

September 2, 2022

## 摘要

当今社会发展迅速，对生产力的要求越来越高，一个企业为了稳固其地位及获得收益，则要寻求合适的原材料供应渠道和制定经济的策略来使自己在成本较低的情况下保证自身的生产链的运转。本文针对生产企业原材料的订购和运输的两个重要过程，运用相关分析、主成分分析、0-1 规划和线性规划等方法在既给情况下为生产企业制定出合理的订购和运输计划。

**针对问题一：**对于问题一，附件的一些数据需要进行预处理。首先利用 pandas 工具对数据缺失值处理。通过比对数据特征分析，我们提取出供货商的供货稳定性、差异性、便宜指数、企业实力以及重大订单接收频次这几项特征值。利用 PCA 对特征值进行主成分分析法，通过加权处理对主成分建立评价打分模型，根据综合主成分分析评价法和秩和比综合评价法分别得到供货商的评分 PCA 得分与 RSR 拟合值，去除极值量纲，将两个评价指标求和得到综合得分。根据得分进行排序，筛选出以 228、360 等 50 家供应商。

**针对问题二：**针对问题二，首先采用时间序列模型，通过对过去 10 个周期的分析，预测了 402 家企业未来 24 周中每周的供货量。后进行特征值处理，得出供货稳定能力、需求关系、损耗率、库存转化率等多项约束条件，构建 0-1 线性规划的最经济订购方案模型，求解能满足生产需求的最少供应商得 24 家。再次使用 0-1 规划模型，得到在满足每周生产需求且预留两周库存量的条件下，订购费用最小的原材料供货方案。考虑一家供应商的材料由一家转运商运输以及转运商有运输能力上限，使用 0-1 背包模型设计转运方案，降低转运过程的损失。最后根据 24 周每周库存量是否合理及转运损失是否减少来评判订购方案和转运方案。

**针对问题三：**运用 0-1 整数规划模型，得到在满足生产需求的条件下成本最小的原材料供货方案，按照前述方法求得相应的供货方案。为了降低转运损耗率，采取尽可能让损耗率小的转运商转运更大体积的货物的方案，以 0-1 背包模型设计转运方案。最后根据未来 24 周预估总成本是否比历史平均成本减少，以及转运损耗率是否降低来评估订购方案和转运方案，最后发现节约了约一定的成本，也降低了损耗率。

**针对问题四：**本问采用多规划模型，首先利用现有的供应商最大供货量和企业的已有产能数据进行估测，利用时间序列的方式来估计企业的产能上限。其次是采用模拟退火算法来进行迭代求解寻优，将产能上限作为新的约束引入混合规划模型，以利润最大为目标，确定企业相应的产能和订货计划。

**关键字：**主成分分析；0-1 规划；模拟退火；蒙特卡洛；多重规划；

# 1 问题背景与重述

## 1.1 问题背景

某生产企业所用的原材料分为 A,B,C 三种类型。该企业的产能要求是 2.82 万立方米/周，其中每生产一立方米产品需要消耗 0.6 立方米的 A 材料，或 0.66 立方米的 B 材料，或 0.72 立方米的 C 材料。

该企业每年生产期为 48 周，需要提前制定 24 周的原材料订购和转运方案，即根据产能要求确定每周的原材料供应商和订货量，再安排将供应商的供货量转运到企业仓库的转运商。

由于供应商的实际供货量可能与订货量有差异，且企业原材料存货量尽量保证满足大于等于两周的生产需求，因此该企业总是全部收购供应商提供的原材料。其中材料 A,B 的购买单价分别是材料 C 的 1.2 倍和 1.1 倍，三类原材料运输和存储的单位费用相同。转运商转运过程中也会损耗原材料。每家转运商每周的运输上限是 6000 立方米，通常一家供应商每周提供的原材料由一家转运商运输。

## 1.2 问题重述

问题一，已知该企业之前 240 周共 402 家供应商的订货量和供货量，以供应商对于保证企业生产的重要性为标准，确定 50 家最重要的供应商。

问题二，在问题一的基础上，计算可以满足企业生产需求的供应商的最小数量。在确定这些供应商的基础上，给出接下来 24 周的最经济的原材料订购方案和损耗最少的转运方案，并分析方案效果。

问题三，制定新的订购和转运方案，以减少运输和存储的成本（多采购材料 A，少采购材料 C）并尽量降低转运材料损耗率，并分析方案效果。

问题四，计算该企业每周可以提高多少产能，并制定接下来 24 周的订购和转运方案。

# 2 问题分析

## 2.1 问题一的分析

第一，根据附件 1 中给出信息分析供货商的供货能力。对于此问题，分析附件一所提供的企业订货量和供应商供货数进行特征提取和量化分析，提取特征。

第二，通过 PCA 对建立的特征值进行主成分分析，通过主成分分析建立打分模型，计算不同供应商的分数，通过分数进行排序选择 50 家最重要的供应商。

## 2.2 问题二的分析

在已有的 50 家供应商中需要确定最小的供应商数来满足自身的产能要求，在保证利益的情况下尽可能地选择比较少的供应商数。

首先，根据要求进行特征提取：供货稳定能力、需求关系、损耗率、库存转化率等多项约束条件。确定 0-1 整数规划的最少供应商模型。其次，在最少供应商的基础上，构建基于线性规划的最经济订购方案模型，求解最经济的订购方案。最后，进行最少损耗的模型建立，求解出最经济且损耗最小，收益最大的订购方案。

## 2.3 问题三的分析

## 2.4 问题四的分析

# 3 符号说明

符号	基本说明
$Goods$	订货量和生产产品
$W_i$	产品种类对应的每立方米产品需消耗对应种类的原材料
$Goods_{i_{std}}$	第 $i$ 个供货商（订货商）对应的第 $j$ 周的供货量的标准差。
$Goods_{i_{mean}}$	中第 $i$ 个供货商（订货商）对应的第 $j$ 周的供货量的均值。
$Goods_{cv_{i_{mean}}}$	即为第 $i$ 个供货商（订货商）对应的第 $j$ 周的变异系数
$Goods_{stab_i}$	为第 $i$ 个供货商（订货商）对应的每一周的稳定指数
$Goods_{diff_i}$	为第 $i$ 个供货商（订货商）对应的每一周的偏移指数
$diff\_metric$	为订货的稳定指数
$vacancy\_rate$	供货商占有率， $data\_orger_{ij}$ 为供货商 $i$ 第 $j$ 周的供货数
$time_{total}$	为总周数
$Default\_rate$	供应商违约率
$Compliance\_rate$	供应商守约率
$important\_freq$	重要订单接收频次
$segmentation_{market\_share}$	为供应商 $i$ 对应的市场份额

# 4 模型假设

1. 该企业在初始一周时的库存量为 0。
2. 企业的库存至少保证两周的生产量，为现周和下周的生产需求量。
3. 在最后一周，仍能保证溢出一周的储备量。
4. 供应商和转运商和企业的合作关系在企业所需供应的时间内保持不变。
5. 为了保证企业的产能，企业的订购量超过实际所需的改为汉字数字。
6. 不存在灾害因素给供应商带来的大规模地断供。
7. 供应商和转运商之间不存在某种关系，其和企业的合作是相互独立的。

# 5 问题一的求解

## 5.1 基于综合主成分分析评价法和秩和比综合评价法的供货能力评价模型的建立

要求建立一套能够反映保障企业生产重要性的数学模型，在此基础上确定 50 家对于该建筑和装饰板材的生产企业的生产有重要性或者重要保障的原材料供应商。由于供应商对于企业的生产的影响是多方面多层次的，为此本文通过建立关于供应商供货能力的评价模型，给附件一中 402 家供应商综合打分，由该得分反映出供货商对于保障企业生产的重要程度，最后挑选出排名前 50 家供应商，确定为对该生产企业最重要的 50 家供应商。

## 5.2 数据处理

我们对于附件 1 的数据，运用 Python 的 pandas 库，找出缺失值。若有缺失值，可删除有缺失值的样本或特征，或对缺失值进行填补。检查数据是否存在缺失值或者 NAN 现象，通过简单的查看可知该题数据不存在缺失，无需进行缺失值处理。

## 5.3 评价指标的选取

参考国内外已有研究用，发现在现有研究中“质量、成本、交货、服务”四个方面是人们评价供应商时关注的重点，通过分析供应商的供应特征以及选择供应商的主要目的，我们深度挖掘附件-中 402 家原材料供应商在近五年内的订货量和供货量的数据，从供货水平、订单完成水平和原材料供应类别这三方面进行细化指标选取，从而选取每周最大供货量、每周平均供货量、近 5 年总供货量、近 5 年订货频次、供货量的标准差、平均综合偏差值、综合偏差值的标准差、单位产品消耗 A、B、C 三类原材料的综合成本 (分为采购和运输及储存两方面) 八个二级的细化指标用以评价供应商的供货能力，具体的细化指标的计算公式如下。

## 5.4 评价指标的量化计算

### 5.4.1 加权处理

由于生产每立方米的材料需要消耗不同立方米的原材料，为了衡量不同原材料的重要性，我们在原材料数上除以产品种类对应的每立方米产品需消耗对应种类的原材料，统一三种材料的量纲。

$$Goods\_weight_{ij} = \frac{Goods_{ij}}{W_{ij}} \quad (1)$$

### 5.4.2 供货订货差异性分析

为了衡量供货与订货之间的稳定性，本文定义了供货订货比，用于衡量供货商的供货水平以及供货稳定性。对于供货订货比: 等于 1，说明供货与订货相等。小于 1，说明供货不足，供货商私信供货能力不足。大于 1，说明供货商提供的货物大于订货，可能会影响到后续的需求。计算各订货量  $D_i$  与供货商生产的产品数  $S_i$  的比率  $R_i$ :

$$R_i = \frac{D_i}{S_i} \quad (2)$$

### 5.4.3 供货商稳定性分析

为了比较供货与订货两个数据离散程度，消除测量尺度和量纲的影响，导致两组数据因统计导致尺度相差太大，或者数据量纲的不同，直接使用标准差来进行比较不合适，而变异系数可以做到这一点，它是原始数据标准差与原始数据平均数的比。CV 没有量纲，这样就可以进行客观比较了。 $Goods\_cv_{i\_mean}$  即为第 i 个供货商（订货商）对应的第 j 周的变异系数，公式定义如下：

$$Goods\_cv_{i\_mean} = \frac{Goods_{i\_std}}{Goods_{i\_mean}} \quad (3)$$

$Goods\_stab_i$  为第 i 个供货商（订货商）对应的每一周的稳定指数：

$$Goods\_stab_i = \frac{1}{Goods\_cv_{i\_mean} + 1} \quad (4)$$

#### 5.4.4 供货偏移指数

为衡量供货商是否能及时供货，我们提出使用供货偏移指数进行建模，判断供货商对于订货商给出订单需求后是否及时供货

$Goods\_diff_i$  为第  $i$  个供货商（订货商）对应的每一周的偏移指数：

$$Goods\_diff = Goods\_stab\_sup_i - Goods\_stab\_ord_i \quad (5)$$

$diff\_metric$  为订货的稳定指数，

$$diff\_metric = \frac{1}{\sum_{n=0}^i \frac{(Goods\_diff_n)^2}{Goods\_diff_n} + 1} \quad (6)$$

#### 5.4.5 供货商实力评判

为了对比供货商之间的实力，我们通过计算供货商提供的总货物数占对应种类市场的份额进行评判比较判断，所占市场份额越大，则说明对应供货商的能力越强。

$$order\_bool_{ij} = \begin{cases} True & data\_order_{ij} = 0 \\ False & data\_order_{ij} \neq 0 \end{cases} \quad (7)$$

$$data\_order\_IdleI_i = \frac{\sum_{n=0}^j order\_bool_{in}}{time_{total}} \quad (8)$$

$$vacancy\_rate_i = 1 - data\_order\_IdleI_i \quad (9)$$

#### 5.4.6 供货商重要订单接收频次

$important\_freq$  被初始化为全 0 的矩阵。

$$max\_order_i = \{data\_order_{ij} | data\_order_{ij} \text{ argsort}(data\_order_{ij}) \wedge 0 \leq j \leq 20\} \quad (10)$$

$$important\_freq_i = important\_freq_{ij} + 1 \text{ s.t. } max\_order_{ij} \text{ max\_order}_i \quad (11)$$

#### 5.4.7 供货商信誉判断

为了判断供货商供货，我们通过计算订货商订货，但是供货商没有供货的天数来判断。

$$data\_order\_bool_{ij} = \begin{cases} True & data\_order_{ij} > 0 \\ False & data\_order_{ij} \leq 0 \end{cases} \quad (12)$$

$$data\_supply\_bool_{ij} = \begin{cases} True & data\_supply_{ij} = 0 \\ False & data\_supply_{ij} \neq 0 \end{cases} \quad (13)$$

$$data\_bool2_{ij} == \begin{cases} True & data\_order\_bool_{ij} \& data\_supply\_bool_{ij} == True \\ False & data\_order\_bool_{ij} \& data\_supply\_bool_{ij} == False \end{cases} \quad (14)$$

$$data\_order2\_bool_{ij} = \begin{cases} True & data\_order_{ij} \neq 0 \\ False & data\_order_{ij} = 0 \end{cases} \quad (15)$$

$$Default\_rate_i = \frac{\sum_{n=0}^i data\_bool_{in}}{\sum_{m=0}^i data\_order2\_bool_{im}} \quad (16)$$

$$Compliance\_rate = 1 - Default\_rate \quad (17)$$

## 5.5 基于综合主成分分析评价法和秩和比综合评价法的供货能力评价模型

### 5.5.1 分析步骤

1. 准备好数据，并且进行同趋势化处理与量纲问题;
2. 确认各指标权重，可使用熵权法、自定义权重、层次分析法（需自行处理，可使用 SPSSPRO 量化分析-AHP）;
3. 计算秩值，根据每一个具体的评价指标按其指标值的大小进行排序，得到秩次 R，用秩次 R 来代替原来的评价指标值;
4. 计算得到 RSR 值和 RSR 值排名;
5. 列出 RSR 的分布表格情况并且得到 Probit 值;
6. 以 Probit 值（累积频率所对应的概率单位）为自变量，以 RSR 值为因变量，计算直线回归方程，拟合所对应的 RSR 估计值;
7. 根据拟合的 RSR 值排序，并且进行分档等级。

### 5.5.2 秩和比综合评价法（RSR）分析结果

使用熵权法进行指标权重计算，下表展示了熵权法的权重计算结果，根据结果对各个指标的权重进行分析。熵权法的权重计算结果显示，订货数量的权重为 21.228%、重要订单接受频次的权重为 23.209%、供货数量的权重为 23.762%、供应商细分市场份额的权重为 23.138%、订货稳定指数的权重为 2.216%、供应商占用率的权重为 2.765%、供应商守约率的权重为 1.405%、供货稳定指数的权重为 1.158%、供货偏移指数的权重为 1.118%，其中指标权重最大值为供货数量（23.762%），最小值为供货偏移指数（1.118%）。

表 1: 熵权法

项	信息熵值 e	信息效用值 d	权重
订货数量	0.643	0.357	21.228
重要订单接受频次	0.61	0.39	23.209
供货数量	0.6	0.4	23.762
供应商细分市场份额	0.611	0.389	23.138
订货稳定指数	0.963	0.037	2.216
供应商占用率	0.953	0.047	2.765
供应商守约率	0.976	0.024	1.405
供货稳定指数	0.981	0.019	1.158
供货偏移指数	0.981	0.019	1.118

进行秩值计算，根据每一个具体的评价指标按其指标值的大小进行排序，得到秩次 R，用秩次 R 来代替原来的评价指标值，根据编秩结果建立各指标的秩次数据矩阵。使用非整秩法进行计算，类似于线性插值的方式对指标值进行编秩，以改进 RSR 法编秩方法的不足，所编秩次与原指标值之间

存在定量的线性对应关系，从而克服了 RSR 法秩次化时易损失原指标值定量信息的缺点。下表展示部分秩值计算结果。

TODO:: 这种太宽的表格怎么办 (T)

索引	X1: 订货数量	R1: 订货数量	X2: 重要订单接受频次	R6: 供应商占用率	X7: 供应商守约率	R7
0	0.000536236	1.215030487	1.00E-04	0.178683732	72.65217652	
1	0.000762555	1.305784611	1.00E-04	0.187645018	76.2456522	
2	0.027258321	11.93058687	0.029260815	0.375976723	151.766666	
3	0.001537944	1.616715407	1.00E-04	0.148061779	60.3727734	
4	0.015002396	7.015960762	1.00E-04	0.627125997	252.4775247	
5	0.000936295	1.375454444	1.00E-04	0.161435272	65.73554407	
6	0.015409314	7.179134844	0.008431647	0.371974055	150.1615962	
7	0.000235238	1.094330557	1.00E-04	0.378469585	152.7663034	
8	0.0015421	1.618382149	1.00E-04	0.060921645	25.4295798	
9	0.001253226	1.502543552	1.00E-04	0.208543322	84.62587205	
10	0.000216188	1.086691321	1.00E-04	0.438579861	176.8705241	
11	0.000760269	1.304867903	1.00E-04	0.193280991	78.50567719	
12	0.000139986	1.056134377	1.00E-04	0.455421798	183.6241408	
13	0.000129318	1.051856405	1.00E-04	0.725164321	291.7908929	

以下表格为部分结果，RSR 的分布是指用概率单位 Probit 表达的值特定的累计频率。其方法为：step 1 将 RSR 值按照从小到大的顺序排列；step 2 列出各组频数；step 3 计算各组累计频数；step 4 确定各组 RSR 的秩次 R 及平均秩次  $R^-$ ；step 5 计算向下累计频率  $R^-/n \times 100\%$ ，最后一项用  $(1 - \frac{1}{4n}) \times 100\%$  修正；step 6 根据累计频率，查询“百分数与概率单位对照表”，求其所对应概率单位 Probit 值；step 7 利用表格中的 RSR 分布值作为自变量，Probit 值作为因变量，进行线性回归，结果如下表格。PS：在编秩过程中进行的是同向趋势化处理，即将负向指标（成本型指标）转化成正向指标（效益型指标），统一对所有指标进行从小到大编秩；

表 3: 部分秩值计算结果

RSR	频率 f	累计频数 $\Sigma f$	评价秩数	评价秩数/ $n \times 100\%$	Probit
0.020844231	1	1	1	0.248756219	2.191359989
0.024206993	1	2	2	0.497512438	2.422446536
0.024803811	1	3	3	0.746268657	2.565815209
0.024872558	1	4	4	0.995024876	2.671781372
0.025293148	1	5	5	1.243781095	2.756671237
0.025377881	1	6	6	1.492537313	2.827934812
0.025529463	1	7	7	1.741293532	2.889622706
0.025803867	1	8	8	1.990049751	2.944191672

0.025924679	1	9	9	2.23880597	2.993248406
0.026052561	1	10	10	2.487562189	3.037903443
0.026059396	1	11	11	2.736318408	3.078957646
0.026165731	1	12	12	2.985074627	3.117008298
0.026277061	1	13	13	3.233830846	3.152513182
0.026356117	1	14	14	3.482587065	3.185831195
0.02639066	1	15	15	3.731343284	3.217249125
0.026417487	1	16	16	3.980099502	3.246999901
0.026651563	1	17	17	4.228855721	3.275275394
0.026657858	1	18	18	4.47761194	3.302235607
0.026673477	1	19	19	4.726368159	3.328015422
0.026962146	1	20	20	4.975124378	3.352729641
0.027170188	1	21	21	5.223880597	3.376476818
0.02722017	1	22	22	5.472636816	3.399342216
0.027249725	1	23	23	5.721393035	3.421400113
0.027298297	1	24	24	5.970149254	3.442715632
0.027316883	1	25	25	6.218905473	3.463346196
0.027355379	1	26	26	6.467661692	3.483342714
0.027437135	1	27	27	6.71641791	3.502750533
0.027514565	1	28	28	6.965174129	3.521610232
0.027730933	1	29	29	7.213930348	3.539958275
0.027753687	1	30	30	7.462686567	3.557827553

下表格展示了本次模型的分析结果，包括模型的标准化系数、t 值、VIF 值、 $R^2$ 、调整  $R^2$  等，用于模型的检验，并分析模型的公式。step 1 线性回归模型要求总体回归系数不为 0，即变量之间存在回归关系。根据 F 检验结果对模型进行检验；step 2  $R^2$  代表曲线回归的拟合程度，越接近 1 效果越好；step 3 VIF 值代表多重共线性严重程度，用于检验模型是否呈现共线性，即解释变量间存在高度相关的关系（VIF 应小于 10 或者 5，严格为 5）；若 VIF 出现 inf，则说明 VIF 值无穷大，建议检查共线性，或者使用岭回归。

从 F 检验的结果分析可以得到，显著性 P 值为 1%，水平呈现显著性，拒绝了回归系数为 0 的原假设，同时模型的拟合优度  $R^2$  为 0.38，模型表现较差，因此模型基本满足要求。对于变量共线性表现，VIF 全部小于 10，因此模型没有多重共线性问题，模型构建良好。对于变量共线性表现，VIF 全部小于 10，因此模型没有多重共线性问题，模型构建良好。模型的公式如下：

$$y = -0.296 + 0.073 \cdot Probit \quad (18)$$

TODO: 这种注放在哪里注：1. \*\*、\* 分别代表 1%、5%、10% 的显著性水平。2. 因变量：RSR。

下图为拟合曲线与真实曲线的对比



表 4: 线性回归分析结果 n=402

	非标准化系数		标准化系数	t	p	VIF	R2	调整 R2	F
	B	标准误	Beta						
常数	-0.296	0.024	-	-12.402	0.000***	-	0.38	0.378	F=245.181 P=0.000***
Probit	0.073	0.005	0.616	15.658	0.000***	1			

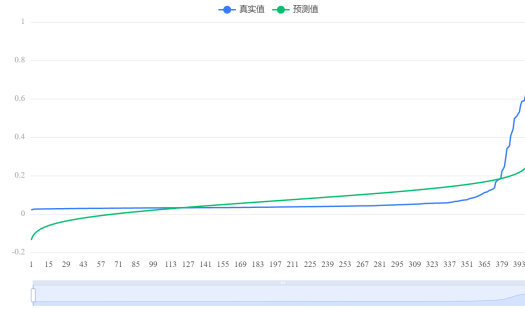


图 1: 拟合效果图

最后，我们得到分档排序临界值表格，尤其是 Probit 临界值对应的 RSR 临界值（拟合值）

我们根据综合主成分分析评价法和秩和比综合评价法分别得到供货商的评分 PCA 得分与 RSR 拟合值，我们将他们进行 min-max 处理，去除两个评分的量纲，将两个评价指标求和得到的综合得分作为最终的评价指标，取评分最高的 Top50 为最终选取的供应商。

表 5: 第二问预测结果

供货商	Probit	RSR 拟合值	PCA 得分	PCA(标准化)	RSR 拟合 (标准化)	综合评分值
228	8.228645179	0.306420133	2.782049173	1	1	2
360	7.808640011	0.275660741	2.601285883	0.950070052	0.930431451	1.880501503
274	7.243328763	0.23425975	2.3525052	0.881352524	0.836794786	1.718147309
328	7.110377294	0.224522951	2.333973533	0.876233757	0.814773057	1.691006814
339	7.172065188	0.229040709	2.278486776	0.860907354	0.824990875	1.68589823
267	6.921042354	0.210656865	2.278209629	0.860830802	0.783412116	1.644242918
130	7.006751594	0.216933845	2.126081771	0.818810455	0.797608768	1.616419224
107	7.577553464	0.258736946	1.713112255	0.704741131	0.892154886	1.596896017
305	6.847486818	0.20526997	2.120079287	0.817152466	0.77122857	1.588381035
193	6.782750875	0.200528985	2.129456061	0.819742493	0.760505879	1.580248372
281	7.055808328	0.220526552	1.905487341	0.757878458	0.805734397	1.563612855
246	6.623523182	0.188867826	1.864908952	0.746670005	0.734131823	1.480801828
351	6.753000099	0.198350165	1.783104178	0.724074111	0.75557804	1.47965215
139	7.434184791	0.248237234	1.283004136	0.585937828	0.868407676	1.454345504
355	6.814168805	0.202829901	1.603297148	0.674408299	0.765709863	1.440118162
150	7.328218628	0.240476723	1.257439062	0.578876312	0.85085572	1.429732032
30	6.557284368	0.184016778	1.704744368	0.702429776	0.723160203	1.425589979
307	6.962096557	0.2136635	1.380909307	0.61298093	0.790212225	1.403193155

329	6.882991702	0.207870197	1.286324272	0.586854907	0.777109506	1.363964413
293	6.298749674	0.165082796	1.584520457	0.669221852	0.680337196	1.349559048
364	6.478389768	0.178238873	1.440791839	0.629521521	0.710092309	1.33961383
142	6.724724606	0.196279388	1.284431535	0.5863321	0.750894561	1.337226661
265	6.216273909	0.159042622	1.48804126	0.642572628	0.666676128	1.309248756
345	6.328298237	0.167246807	1.313626842	0.594396349	0.685231542	1.279627891
39	6.460041725	0.17689514	1.18975455	0.560180679	0.707053187	1.267233866
283	6.536653804	0.182505883	1.081315281	0.530227877	0.71974301	1.249970887
366	6.391152774	0.171850006	1.086761145	0.531732119	0.695642602	1.227374721
122	6.165587919	0.155330596	1.162793881	0.552733676	0.658280635	1.211014311
217	6.242830371	0.160987505	1.108545615	0.537749366	0.671074872	1.208824238
363	6.42475216	0.174310682	0.967951043	0.498914713	0.701207916	1.200122628
243	6.256443485	0.161984471	1.051499232	0.521992168	0.673329711	1.195321879
79	6.284391587	0.164031271	1.006025901	0.509431647	0.677958961	1.187390608
66	6.061292014	0.147692408	1.050982553	0.521849452	0.641005337	1.16285479
138	6.671984578	0.192416933	0.656463001	0.412876327	0.742158842	1.15503517
188	6.129453754	0.152684284	0.972738958	0.500237218	0.652295468	1.152532686
212	6.05040415	0.146895027	0.954365241	0.495162079	0.6392019	1.134363979
350	5.810051726	0.129292638	1.084013747	0.53097324	0.599390558	1.130363798
75	6.072307165	0.14849911	0.85290527	0.46713708	0.642829856	1.109966936
396	5.759152739	0.125565012	1.029222365	0.515838912	0.590959782	1.106798694
4	6.028993803	0.145327024	0.866036047	0.470764028	0.635655546	1.106419574
394	6.600657784	0.187193262	0.488672451	0.366529674	0.73034446	1.096874135
179	5.734430232	0.123754441	0.981421551	0.5026355	0.586864812	1.089500312
200	6.697764393	0.194304937	0.40293102	0.342846408	0.746428943	1.089275351
347	6.647270359	0.190606969	0.388700859	0.338915791	0.738065245	1.076981036
54	6.374926512	0.170661664	0.537650657	0.380058302	0.692954928	1.07301323
45	5.987529589	0.142290361	0.759295529	0.44128045	0.628787522	1.070067972
6	6.153375681	0.154436224	0.627358189	0.404837074	0.656257833	1.061094907
365	5.726288885	0.123158204	0.873465466	0.472816162	0.585516302	1.058332464
97	6.039639404	0.146106663	0.681039986	0.419664916	0.637418856	1.057083772

## 6 问题二的求解

题目要求“参考问题 1”，确定至少选择多少家供应商供应原材料才可能满足生产的需求。因此，本文首先将供应商压缩为排名前 50 的供应商，继而针对这 50 家供应商，建立 0-1 整数规划模型，以满足生产需要的最少的供应商数目为目标函数，从而求解出选用哪几家供应商供应原材料，并且这些供应商的总数为多少。

## 6.1 目标函数的构建

为尽可能选择少的供应商以满足生产的需要，本文从 402 家供应商中选出供货能力排名前 50 家供应商作为研究对象。所以，本文构建的目标函数为：

$$\sum_{i=1}^{50} x_{ij} \quad (19)$$

其中  $x_{ij}$  代表第  $i$  个供应商在第  $j$  周是否向该企业供货。若  $x_{ij}$  为 1 则代表供货，反之则代表没有供货。

### 6.1.1 约束条件的构建

#### 1. 稳定供货能力

为了求解满足企业生产需要的最少的供应商数量，本文当中假设每个供应商的供货量达到自己的最大供货水平。但是从实际情况来看，对于一家供应商来说，24 周每周向企业提供的供应量不可能都是历史最大供货量，供货量也可能有起伏。因此稳定的供货能力不能简单的视为最大的供货量，实际中稳定供货能力应小于或等于历史最大供货量。定义供货系数为  $B, B \in (0, 1)$ ，使得：

$$\text{稳定的供货能力 } S_{ij} = B \cdot \text{最大供货能力} \quad (20)$$

#### 2. 供给需求关系

由于在第  $j$  周所有供货商的供货经过转运到达企业后应满足企业本周的生产需要和库存需要，因此存在供给需求的等式关系，即该生产企业在第  $j$  周实际的原材料接收量与第  $j-1$  周企业仓库中剩余的原材料库存量之和必须大于等于企业本周的产能所需原材料量和两周生产需求的原材料库存量。为了便于计算，我们将等式两边的原材料接收量与原材料库存量均转化为对应的企业产能，则供给需求关系式如下：

$$\sum_{i=1}^{50} x_{ij} S_{ij} (1 - L_{ij}) f_{type}(i) + R_{j-1} = P + S_{2w} \quad (21)$$

其中， $f_{type}(i)$  为对原材料加工的转换函数，是计算将不同种类的原材料转换成成品的函数。 $S_{ij}$  表示第  $i$  个供货商在第  $j$  周给该企业的供应量； $L_{ij}$  表示第  $i$  个供货商在第  $j$  周给该企业的供货过程中的损耗率； $R_j$  表示在第  $j$  周后剩余库存转换为相应产能的产量， $P$  表示企业的每周产能， $S_{2w}$  表示满足企业两周用量的原材料库存数量。

**3. 损耗率**在考虑第  $i$  个供货商在第  $j$  周发出的供货量在转运过程中的损耗率  $L_{ij}$  时，由于无法确定转运公司的选择，所以采用附件二中关于转运商 240 周的损耗率的平均值为损耗率，统计出以 24 周为周期每周的平均损耗率，计算公式如下：

$$L_{ij} = L = \frac{\sum_{k=1}^8 L_{jk}}{8} \quad (22)$$

这种情况下第  $i$  个供货和第 1 个在第  $j$  周的损耗率相同，则：

$$L_{ij} = L_j = \frac{\sum_{k=1}^8 L_{jk}}{8} \quad (23)$$

#### 4. 库存转换为产能

根据题意，该企业要尽可能保持不少于两周生产需求的原材料库存量，因此，有关系式如下：

$$S_{2w} = 2P \quad (24)$$

其中  $P$  为库存,  $S_{2w}$  为产能。

### 5. 库存剩余量的计算迭代公式

第  $j$  周的原材料库存剩余量  $R_j$ , 应等于第  $j$  周的原材料接收量与第  $j-1$  周的原材料库存剩余量之和再减去第  $j$  周的产能所耗的原材料。本文将等式两边的原材料接收量与原材料库存剩余量均转化为相应的产能, 所以库存剩余量的迭代关系式如下:

$$\begin{cases} R_j = \sum_{i=1}^{50} x_{ij} S_{ij} (1 - L_{ij}) f_{type}(i) + R_{j-1} - P \\ R_{j-1} = \sum_{i=1}^{50} x_{i(j-1)} S_{i(j-1)} (1 - L_{i(j-1)}) f_{type}(i) + R_{j-2} - P \\ \vdots \\ R_1 = \sum_{i=1}^{50} x_{i1} S_{i1} (1 - L_{i1}) f_{type}(i) + R_0 - P \end{cases} \quad (25)$$

### 6. 初始库存量

在知道库存剩余量的迭代关系后, 确定初始的库存剩余量  $R_0$  至关重要。考虑到企业依旧持有库存, 所以在上一个为期 24 周的生产结束时企业不会将仓库中所有的库存量全部生产用尽, 因此不能简单的认为  $R_0 = 0$ 。

折线图

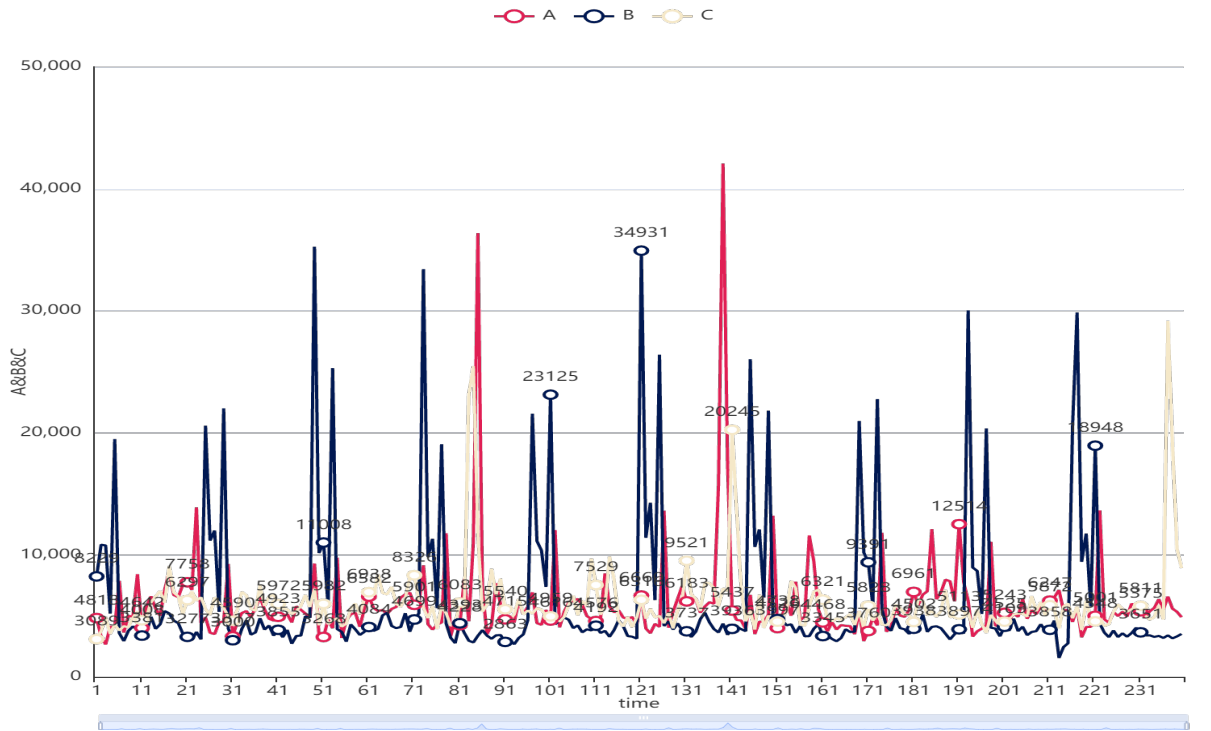


图 2: 初始库存量

根据上图的订货量曲线不难发现订货量大致呈现随周期变化的规律, 且周期长度为 24 周。所以在以 24 周为周期的订购计划的开始, 初始库存剩余量大致可用以往周期的初始库存剩余量来计算。因此不妨设  $R_0 = S_{2w}$ 。

## 6.2 基于 0-1 整数规划的最少供应商模型

在满足企业的生产需求的情况下，尽可能用最少的供应商向该企业提供生产的原材料，结合已知条件和假设，基于 0-1 整数规划的最少供应商模型确立为：

$$\sum_{i=1}^{50} x_{ij} \quad (26)$$

$$\text{s.t.} \begin{cases} \sum_{i=1}^{50} x_{ij} S_{ij} (1 - L_{ij}) f_{type}(i) + R_{j-1} = P + S_{2w} \\ R_{j-1} = \sum_{i=1}^{50} x_{i(j-1)} S_{i(j-1)} (1 - L_{i(j-1)}) f_{type}(i) + R_{j-2} - P \\ L_{ij} = \bar{L} \\ S_{ij} = B \cdot \text{最大供货能力} \\ R_0 = S_{2w} \\ S_{2w} = 2P \end{cases}$$

需要注意：由于迭代，该 0-1 整数规划模型对于每个  $j$  (即周) 都是一个全新的规划。

## 6.3 问题二最少供应商模型求解

在满足生产需求的情况下的得到最少的供应商数为 24 个，结果如下：

S229	S308	S356	S247
S361	S282	S268	S284
S151	S340	S306	S055
S108	S275	S194	S201
S330	S329	S143	S003
S229	S131	S352	S037

## 6.4 基于线性规划的最经济订购方案模型的建立

为了制定未来 24 周每周“最经济”的原材料订购方案，本文针对上述模型求解出的 24 家原材料供应商，建立线性规划模型，在必须满足企业的生产需求的约束条件下，以 24 周该生产企业需要给原材料供应商的采购费用和给物流公司 (即转运商) 的运输费用以及在仓库储存费用之和达到最小作为目标函数，从而确定出 24 家原材料供应商每周接到的订货量 (总共 24 周)。

### 6.4.1 目标函数的构建

由于制定订购方案需要明确该生产企业需要订购的原材料供应商以及向这些供应商每周 (总共 24 周) 订购的订货量，因此我们选择以第  $i$  家供应商在第  $j$  周接到的订货量为自变量，即  $O_{ij}$ 。根据题目要求，本文此处的供应商已经确定为上述 0-1 整数规划模型所求解得到的 24 家供应商，因此  $i=1,2,\dots,24, j=1,2,\dots,24$ 。以 240 周内所有供应商提供的总供货量在采购和运输以及储存该三方面花费的成本最小构造目标函数，表达式如下：

$$\min \sum_{i=1}^{24} \sum_{j=1}^{24} O_{ij} (1 - r_{os_{ij}}) (r_i p + c) \quad (27)$$

### 6.4.2 约束条件的构建

由于订购方案是为了满足企业的正常生产需求，因此本文此处的约束条件与基于 0-1 整数规划的最少供应商模型相同，只是变量的表示发生了变化，因此此处不再多余赘述约束条件的构建。

### 6.4.3 基于线性规划的最经济订购方案模型

为了制定未来 24 周每周最经济的订购方案，即要求订购的原材料总量在采购、运输及储存三方面的成本花费为最小，在制定方案的同时也需要保证该订购方案满足企业的生产需求，结合已知条件和假设，该规划模型需要满足以下约束：

1. 该生产企业对一周的原材料接收量和上一周剩余的原材料库存量的总和不少于满足两周生产需要的原材料库存量与一周的产能需要的原材料的总和。
2. 一周的原材料库存量是来源于该周的原材料接收量与上一周剩余的原材料库存量的总和中除去该周的产能需要的原材料量后剩余的数量。
3. 该生产企业第一周的库存量为两倍的产能所需原材料。
4. 该企业尽可能保持不少于满足两周生产需求的原材料库存量。
5. 为了计算方便，本文将原材料的接收量和库存量转换为相应的等量产能。

因此，基于线性规划的最经济订购方案模型确立为：

$$\min \sum_{i=1}^{24} \sum_{j=1}^{24} O_{ij}(1 - r_{os_{ij}})(r_i r_1 + r_2) \quad (28)$$

$$\text{s.t.} \begin{cases} \sum_{i=1}^{24} x_{ij} O_{ij}(1 - r_{os_{ij}}(1 - L_{ij}))f_{type}(i) + R_{i-1} = P + S_{2w} \\ R_{j-1} = \sum_{i=1}^{50} x_{i(j-1)} O_{i(j-1)}(1 - r_{os_{ij}}(1 - L_{i(j-1)}))f_{type}(i) + R_{j-2} - P_i \\ L_{ij} = \bar{L} \\ r_{os_{ij}} = r_{os_{ij}} \\ R_0 = S_{2w} \\ S_{2w} = 2P \end{cases}$$

其中， $r_{os_{ij}}$  表示以 240 周为时间总长，24 周为一个单位时间长度得到的第  $i$  家供应商在第  $j$  周的平均订购偏差率， $r_1$  是第  $i$  个供应商供应的原材料的采购单价相对 C 类原材料采购单价的比值，其可能取值为  $\{1.2, 1.1, 1\}$ ， $P$  表示 C 类原材料的采购单价， $C$  表示原材料运输和储存的单位费用。

## 6.5 基于线性规划的最经济订购方案模型的求解

### 方案实施效果分析：

根据本文模型建立的原理，每周 8 家转运商总转运量应大致等于该周向 24 家供应商订货总数，输出结果显示求解结果与预期吻合。此外，可知 24 周平均每立方米原材料需单位成本如下，可见订购方案较为经济。

1	2	3	4	5	6	7	8	9	10	11	12
1.60	1.61	1.57	1.57	1.62	1.56	1.57	1.58	1.58	1.62	1.52	1.55
13	14	15	16	17	18	19	20	21	22	23	24
1.65	1.65	1.57	1.57	1.65	1.57	1.57	1.65	1.57	1.57	1.65	1.60

## 6.6 基于线性规划的最少转运损耗方案模型的建立

根据上述基于线性规划的最经济订购方案模型,可以得到 24 家供应商每周接到的订购量,据此来制定每周内每家供应商的供货量分配给哪一家/几家转运商来运输,并以每家转运商的运输能力为 6000 立方米/周作为约束条件之一,建立线性规划模型,以 8 家转运商在 24 周的转运过程中的总损耗量最少为目标函数,从而确定出每周每家转运商转运哪一家/几家供货商的供货量。

由于过去的转运损耗不会影响本周的转运损耗,所以要使转运损耗最小只需要使每周的转运损耗都最小,为了简便计算可以一周为单位考虑。假设第  $j(j=1,2,\dots,24)$  周第  $i(i=1,2,\dots,24)$  家供货商由第  $k(k=1,2,\dots,8)$  家转运公司转运的原材料量为  $T_{ik}$ ,该转运公司的转运损耗率为  $L_k$ ,则第  $j$  周所有货物的总损耗为  $\sum_{i=1}^{24} \sum_{k=1}^8 T_{ik} L_k$ ,所以为了使得转运损耗最小,有目标函数:

$$\min \sum_{i=1}^{24} \sum_{k=1}^8 T_{ik} L_k, \text{ for } j = 1, 2, \dots, 24 \quad (29)$$

### 6.6.1 约束条件的建立

#### 1. 转运损失率

由于同一个转运公司不同时间转运的损耗率不同,且由上文可知,整个订购运送过程具有周期性,近 5 年共有 10 个周期,周期为 24 周,故转运公司  $k$  在第  $j$  周的转运损失率可以用以往 10 个周期中第  $j$  周转运损失率的均值  $L(k,j)$  表示,计算关系式如下:

$$L_k = L(k,j) \quad (30)$$

#### 2. 承接量

根据题意每家转运商的运输能力为 6000 立方米/周,所以第  $k$  家转运公司在 24 周内给第  $i$  家企业转运原材料的总量不大于 6000 (单位: 立方米),不等式如下:

$$\sum_{i=1}^{24} T_{ik} \leq 6000, \text{ for } k = 1, 2, \dots, 8 \quad (31)$$

#### 3. 供货量

若第  $i$  家供货商由第  $k$  家转运公司转运的原材料量为  $T_{ik}$ ,则该供货商由各个转运商转运的原材料量的总和应该为该供货商的总供货量,即:

$$\sum_{k=1}^8 T_{ik} = S_i, \text{ for } i = 1, 2, \dots, 24 \quad (32)$$

其中,  $S_i$  为第  $i$  家供货商在第  $j$  天的给定的总供货量。

### 6.6.2 基于线性规划的最经济转运方案模型

基于线性规划的最经济转运方案模型确立为:

$$\begin{aligned} & \min \sum_{i=1}^{24} \sum_{k=1}^8 T_{ik} L_k \\ \text{s.t. } & \begin{cases} L_k = \bar{L}(k,j) \\ \sum_{i=1}^{24} T_{ik} \leq 6000, \text{ for } k = 1, 2, \dots, 8 \\ \sum_{k=1}^8 T_{ik} = S_i, \text{ for } i = 1, 2, \dots, 24 \end{cases} \end{aligned} \quad (33)$$

该线性规划模型只针对求解一周的转运方案,为了获得 24 周的转运方案,则需要重复使用该模型,其中  $T_{ik}, L_k, S_i$  都会随着周数的改变而改变。但因为两个约束条件分别是关于 24 周每周的约束条件和 8 家转运商每家的约束条件,所以该模型的可解性是一定被保证的。

## 6.7 基于线性规划的最少损耗转运方案模型的求解

方案实施效果分析:

24 周平均每周转运损耗率如下, 可见转运方案转运效率良好:

1	2	3	4	5	6
0.1397	0.2122	0.1717	0.1859	0.2003	0.2567
7	8	9	10	11	12
0.1189	0.0280	0.1960	0.2439	0.1233	0.2115
13	14	15	16	17	18
0.2936	0.3483	0.3442	0.1412	0.4659	0.4881
19	20	21	22	23	24
0.1621	0.1621	0.1621	0.1621	0.1621	0.1621

## 7 问题三的求解



## 8 附录

### 8.1 代码

#### 8.1.1 第一问代码

```
1  import numpy as np
2  import pandas as pd
3  from copy import deepcopy
4
5  # 数据读取
6  data1 = np.array(pd.read_excel('./excel/附件1近5年402家供应商的
    相关数据.xlsx', sheet_name=0).fillna(0))
7  data2 = np.array(pd.read_excel('./excel/附件1近5年402家供应商的
    相关数据.xlsx', sheet_name=4).fillna(0))
8
9  data_order = data1[:, 2:]
10 data_supply = data2[:, 2:]
11 data_supplier = data1[:, 0:2]
12
13 # 计算各供应商订货量/供货量能够生产的产品数
14 data_order2 = np.zeros(data_order.shape)
15 for i in range(data_order.shape[0]):
16     if data_supplier[i, 1] == 'A':
17         data_order2[i] = data_order[i] * (1 / 0.6)
18     elif data_supplier[i, 1] == 'B':
19         data_order2[i] = data_order[i] * (1 / 0.66)
20     else:
21         data_order2[i] = data_order[i] * (1 / 0.72)
22 # data_order2= data_order2.sum() / 240
23
24 data_supply2 = np.zeros(data_supply.shape)
25 for i in range(data_supply.shape[0]):
26     if data_supplier[i, 1] == 'A':
27         data_supply2[i] = data_supply[i] * (1 / 0.6)
28     elif data_supplier[i, 1] == 'B':
29         data_supply2[i] = data_supply[i] * (1 / 0.66)
30     else:
31         data_supply2[i] = data_supply[i] * (1 / 0.72)
32
33 # 供货数量
34 supply_num = data_supply2.sum(axis=1)
35 # 供货稳定指数 1/(供货量变异系数+1)
```

```

36 supply_stability = np.array(
37     [np.std(data_supply2[k][data_order2[k] != 0]) / np.mean(
38         data_supply2[k][data_order2[k] != 0])
39     for k in range(402)])
39 supply_stability2 = 1 / (supply_stability + 1)
40 # 供货偏移指数 1/(供货量与订货量差的平方平均数+1)
41 diff = data_supply2 - data_order2
42 supply_shift = []
43 for k in range(402):
44     shift_i = np.mean(np.square(diff[k][data_order2[k] != 0] /
45         data_order2[k][data_order2[k] != 0]))
46     supply_shift.append(shift_i)
47 supply_shift = np.array(supply_shift)
48 supply_shift2 = 1 / (supply_shift + 1)
49 # 订货数量
50 order_num = data_order2.sum(axis=1)
51 # 订货稳定指数 1/(订货量的变异系数+1)
52 order_stability = np.array(
53     [np.std(data_order2[k][data_order2[k] != 0]) / np.mean(
54         data_order2[k][data_order2[k] != 0])
55     for k in range(402)])
56 order_stability2 = 1 / (order_stability + 1)
57 # 供应商占用率 1-闲置率
58 vacancy_rate = (data_order == 0).sum(axis=1) / 240
59 vacancy_rate2 = 1 - vacancy_rate
60 # 供应商守约率 1-违约
61 default_rate = ((data_order > 0) & (data_supply == 0)).sum(axis=1) / (data_order != 0).sum(axis=1)
62 default_rate2 = 1 - default_rate
63 # 重要订单接收频次
64 important_freq = np.zeros(402)
65 for i in range(data_order2.shape[1]):
66     index = np.argsort(data_order2[:, i])[::-1][0:20]
67     important_freq[index] += 1
68 # 供应商细分市场份额
69 all = np.zeros(402)
70 for i in range(len(all)):
71     if data_supplier[i, 1] == 'A':
72         all[i] = np.sum(data_supply2[data_supplier[:, 1] == "A", :])
73     elif data_supplier[i, 1] == 'B':
74         all[i] = np.sum(data_supply2[data_supplier[:, 1] == "B", :])

```

```

73         else:
74             all[i] = np.sum(data_supply2[data_supplier[:, 1] == "C",
75                                     :])
76
77             segmentation_market_share = data_supply2.sum(axis=1) / all
78
79             # 整合各特征
80             data_feature = np.vstack([supply_num, supply_stability2,
81                                     supply_shift2, order_num, order_stability2, important_freq,
82                                     segmentation_market_share,
83                                     vacancy_rate2, default_rate2]).T
84
85             # 特征数据输出
86             data_out = pd.DataFrame(data_feature)
87             data_out.columns = ['供货数量', '供货稳定指数', '供货偏移指数',
88                               '订货数量', '订货稳定指数', '重要订单接受频次',
89                               '供应商细分市场份额', '供应商占用率', '供应
90                               商守约率']
91             data_out.to_excel('第1问参数数值.xlsx')
92
93             # 特征归一化处理
94             from sklearn.preprocessing import MinMaxScaler
95
96             feature_scaled = deepcopy(data_feature)
97             feature_scaled[:, [0, 3, 5]] = MinMaxScaler().fit_transform(
98                 feature_scaled[:, [0, 3, 5]])
99
100             # 主成分加权
101             from sklearn.decomposition import PCA
102
103             pca = PCA(svd_solver='full')
104             feature_pca = pca.fit_transform(feature_scaled)
105             variance = pca.explained_variance_ratio_.reshape(-1, 1)
106             supplier_score = feature_pca @ variance
107             components = pca.components_
108             pd.DataFrame(supplier_score).to_excel('供应商得分.xlsx') # 输出
109             # 402家企业加权得分
110             pd.DataFrame(np.hstack((variance, components))).to_excel('主成分
111             .xlsx') # 输出主成分的参数
112
113             # 计算各企业最大供应量
114             max_offer = data_supply2.max(axis=1).reshape(-1, 1)
115             a = np.hstack((supplier_score, max_offer))
116             a = a[np.argsort(a[:, 0])[:, -1]] # a从大到小排序

```

```
108 m = np.array([a[:i, 1].sum() for i in range(50)])
```

### 8.1.2 问题二代码

```
1 import pandas as pd
2 import numpy as np
3 from matplotlib import pyplot as plt
4 import seaborn as sns
5 import numpy as np
6 import cvxpy as cp
7 from scipy import optimize as op
8
9 pred_result = pd.read_csv('TSA_decided.csv')
10 pred_result['id'] = pred_result['Unnamed: 0'] + 1
11
12 pred_result.set_index('id', inplace=True)
13 pred_result.drop('Unnamed: 0', axis=1, inplace=True)
14 candidates = pd.read_csv('n_sup_decided.csv')
15 candidates['n_id'] = [int(x[1:]) for x in candidates['id'].values]
16 selected = candidates.loc[range(50)]
17
18 selected.supply_amount.values.sum() / 5
19
20 avg_need_eqC = (2.82 * 10 ** 4) * 0.72
21 '每周生产需求(eqC): ' + str((2.82 * 10 ** 4) * 0.72)
22
23 trans = pd.read_csv('trans.csv')
24 def replace0(x):
25     if x == 0:
26         return np.nan
27     else:
28         return x
29
30 notnull = 8 * 240 - trans.isnull().sum().sum()
31 avg_los = (trans.sum().values[1:].sum() / notnull) * 0.01
32
33 '有运输时的平均货损率: ' + str(avg_los)
34
35 selected_pred = pred_result.loc[selected['n_id'].values]
36
37 month_sup_50 = selected_pred.sum(axis=0)
38 need_array_eqC = np.array([avg_need_eqC for i in range(24)])
```

```

39 plt.figure(figsize=[30,4])
40 plt.xticks(range(1,25))
41
42
43 plt.bar(x=range(1,25), height=month_sup_50, label='month_sup_50',
         color='darkred')
44 plt.bar(x=range(1,25), height=need_array_eqC, label='need_array_eqC',
         color='darkgray', alpha=0.7)
45
46 plt.legend()
47 plt.title('50_Need_&_Supply_Barplot')
48 plt.savefig('50_Need_&_Supply_Barplot.jpg')
49
50 selected_pred.sort_index(inplace=True)
51 selected_pred.to_csv('for_matlab.csv')
52 selected_pred = selected_pred.T
53 for i in range(len(selected_pred)):
54     if i<=22:
55         selected_pred.iloc[i+1] = selected_pred.iloc[i+1] +
56             selected_pred.iloc[i]
57
58     c = np.array([1 for i in range(50)])
59
60     a = np.array(selected_pred)
61
62     b = np.array([avg_need_eqC*i for i in range(1,25)])
63
64     x = cp.Variable(50, integer=True)
65
66     obj = cp.Minimize(c*x)
67
68     cons = [a*x>=b, x>=0, x<=1]
69
70     prob = cp.Problem(obj, cons)
71
72     prob.solve(solver='GLPK_MI', verbose=True)
73
74     order_eqC = pd.read_csv('supply_eqC.csv')
75
76
77 order_eqC['id'] = [int(x[1:]) for x in order_eqC['供应商ID'].values]
78 selected_eqC = order_eqC.set_index('id').loc[opt_res1_df.id]

```

