

所属类别	2022 年“华数杯”全国大学生数学建模竞赛	参赛编号
本科组		CM2205029

插层熔喷非织造材料的性能控制研究

摘要

熔喷非织造技术是高效生产和加工产业用纺织品的重要手段, 工艺流程短、生产效率高, 其产品具有纤维超细、比表面积大、孔隙率高等特点, 在医卫防护、保暖隔热、吸音降噪等领域具有独特的优势. 但是, 由于熔喷非织造材料纤维非常细, 在使用过程中经常因为压缩回弹性差而导致其性能得不到保障. 因此, 科学家们创造出插层熔喷法, 即通过在聚丙烯 (PP) 熔喷制备过程中将涤纶 (PET) 短纤等纤维插入熔喷纤维流, 制备出了“Z 型”结构的插层熔喷非织造材料. 而通过分别建立工艺参数与结构变量、结构变量和产品性能之间的关系模型, 有助于为产品性能调控机制的建立提供一定的理论基础, 更好实现产品落地批量生产。

针对问题一, 我们将其分为两个小问进行研究. 对于第一个小问, 我们选择以定性的方式研究插层后结构变量、产品性能的变化规律. 通过对于数据预处理后画出散点图, 我们发现数据呈现饼状分布, 于是采用定性描述的方式阐述了变化规律. 其中, 除了过滤阻力与插层率显示的是负相关, 其余五个结构变量或工艺参数 (厚度, 孔隙率, 压缩回弹性率, 过滤效率, 透气性) 都显示为正相关. 对于第二问, 我们采取了**灰色关联度分析法**, 对于每个结构变量、产品性能都一次计算了这个变量与其他变量的关联度. 从结果来看, 不仅插层率对结构变量、产品性能有较大影响, 结构变量、产品性能之间也存在相互影响. 这也影响了我们之后问题解决中方法的选用。

针对问题二, 由于观察到结构变量、产品性能之间的灰色关联度较大, 于是放弃将变量割裂开分析的多元线性回归的方法, 采用**偏最小二乘回归**的方法, 探究工艺参数和结构变量之间的关系. 通过对于数据预处理、求相关系数矩阵等一系列操作, 我们绘制出了回归系数直方图并拟合出了相应的直线方程, 并根据直线方程成功预测出了在一定工艺参数下, 结构变量的具体数值。

针对问题三, 我们同样采用了偏最小二乘回归的方法, 探究工艺参数与产品性能之间的关系, 以及结构变量之间和产品性能之间的关系. 方法选用的合理性与问题二类似. 对于第一小问, 我们绘制了回归系数的直方图, 方便我们直观迅速地观察自变量再解释因变量时的边际作用. 另外, 比较特别的一点是, 压缩回弹性率对于不同的因变量呈现非常不同的解释能力. 这种复杂的, 局部的相关性在偏最小二乘回归模型中得到有限的展现, 体现了我们模型的优越性. 同时, 我们还绘制了预测图来**检验模型的精度**. 在预测图上如果所有的点在对角线附近分布则说明模型拟合得很好。

针对问题四, 为了满足产品生产需要兼顾各方面的条件和要求, 保证工艺参数的限制和应用的要求, 同时追求较高的过滤效率和较小的过滤阻力的目标, 我们建立了基于过滤效率和过滤阻力的**多目标优化模型**, 通过调整工艺参数, 在保证约束条件的前提下, 使得过滤效率尽量的高的同时力求过滤阻力尽量的小, 得到最优的工艺参数, 进而优化产品性能。

关键词：灰色关联度分析 偏最小二乘回归分析 多元目标优化模型 精确度检验

1 问题提出

1.1 背景分析

熔喷非织造材料工艺属于聚合物挤压成网法的一种，起源于 20 世纪 50 年代初期。60 年代中期至 70 年代，美国 ESSO 公司(今埃克森公司)对该方法进行了改进，并转为民用，使得熔喷技术迅速发展。从 80 年代开始，由于熔喷非织造材料在过滤、阻菌、吸附、保暖、防水等方面的优异性能，被广泛应用于口罩、过滤材料、保暖材料、手术服、防护服、卫生巾、纸尿裤等产品。

熔喷非织造工艺是一个非稳态纺丝过程，会受到诸多因素的影响。现在科学家们创造出插层熔喷法，即通过在聚丙烯(PP)熔喷制备过程中将涤纶(PET)短纤等纤维插入熔喷纤维流，制备出了“Z 型”结构的插层熔喷非织造材料。本文旨在别建立工艺参数与结构变量、结构变量和产品性能之间的关系模型，助力产品落地并实现批量生产。

1.2 问题重述

某机构针对插层熔喷非织造材料进行了一系列研究，主要研究了各种工艺参数和结构变量对于最终产品性能的影响，结果在“C 题数据.xlsx”中给出。本题希望通过数学建模解决以下问题：

1. 建立模型研究插层后结构变量、产品性能的变化规律，并分析插层率对于这些变化有多大程度的影响。
2. 建立模型研究工艺参数和结构变量之间的关系，并对表中 8 个工艺参数条件下的结构变量进行预测。
3. 建立模型研究结构变量与产品性能的关系，以及结构变量之间、产品性能之间的关系，并指出当工艺参数为多少时，产品的过滤效率将会达到最高。
4. 根据实际情况和参数限制，为了使得得过滤效率尽量的高的同时力求过滤阻力尽量的小，工艺参数应该如何设定？

2 问题分析

2.1 问题一的分析

问题一要求根据题目附件所给出的数据，研究建立模型研究插层后结构变量、产品性能的变化规律，并分析插层率对于这些变化有多大程度的影响。在第一小问中，通过观测题目给出的数据，发现同一组中是否有一定插层率对于结构变量有一定影响，而题目正是要求我们探究这种影响的规律性。于是我们确定利用皮尔逊系数判断数据是否存在相关性。通过以插层率百分比作为自变量，结构变量和产品性能的变化百分比作为因变量和对于每一个因变量的 25 组数据进行折线图的绘制和非线性及线性拟合，绘制出 6 个百分比图。之后我们使用灰色关联度分析，分析插层率对于每一个变量的影响，结果将在正文中有所阐述。

2.2 问题二的分析

本题为多自变量多因变量的回归问题。在题目中提示我们参数之间会互相影响，也就是自变量之间，因变量之间存在相关性。如果直接使用多元回归模型就无法建

立这些隐含的相关性。相关性也在我们在上述步骤中计算得出的相关系数矩阵中得到印证。此外，我们的自变量为结构变量的 3 个指标，指标数量很少，如果筛选在释响应时无用的变量然后去除是不可行。因此我们采用偏最小二乘回归 (PLS) 的方法，PLS 可以集主成分分析、典型相关分析和多元线性回归分析的有点于一身，实现存在相关性的多自变量和多因变量的建模。

2.3 问题三的分析

问题三要求建立模型研究结构变量与产品性能的关系，以及结构变量之间、产品性能之间的关系，并指出当工艺参数为多少时，产品的过滤效率将会达到最高。由于面临自变量和因变量之间存在关联的问题，我们仍然采用偏最小二乘回归分析。同时，我们还需要通过绘制预测图来检验模型的精度。在预测图上，如果所有的点在对角线附近分布则说明模型拟合得很好。

2.4 问题四的分析

问题四要求满足产品生产需要兼顾各方面的条件和要求，保证工艺参数的限制和应用的要求，同时追求较高的过滤效率和较小的过滤阻力。我们建立了基于过滤效率和过滤阻力的多目标优化模型，通过调整工艺参数，在保证约束条件的前提下，使得过滤效率尽量的高的同时力求过滤阻力尽量的小，得到最优的工艺参数，进而优化产品性能。我们将会使用 NASG-II 绘制 Pareto 前沿并选取最优点，最后得出相应的结论。

3 符号说明

符号	符号含义及说明
插层率	α
接受距离	x_1
热空气速度	x_2
厚度	y_1
孔隙率	y_2
压缩回弹性	y_3
过滤阻力	w_1
过滤效率	w_2

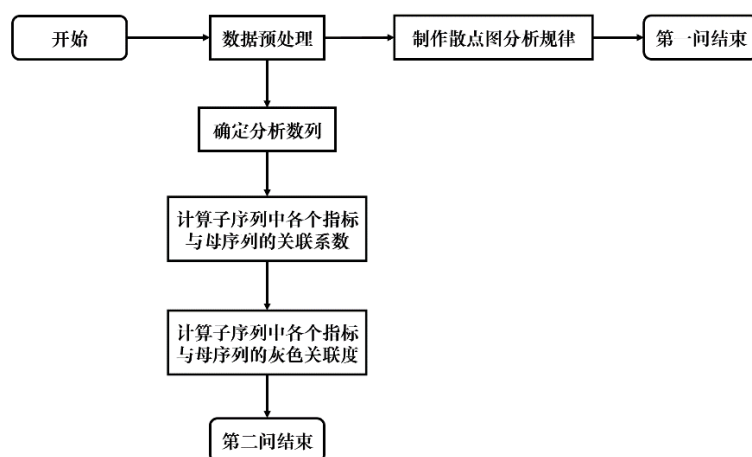
透气性 w_3

表一 符号说明

4 模型建立

4.1 问题一的求解与分析

根据题意，将题目分为两个小问。第一小问通过对于每一组的某一个自变量作差并归一化，得到对于某一个结构变量或工艺参数之于不同插层率的变化数据。通过数据可视化以及线性或非线性的拟合可以分析出相应的变化规律。对于第二小问，我们采取灰色关联度分析，可以得到插层率对于某一个结构变量或工艺参数的影响大小。



4.1.1 数据预处理

我们选取其中一组数据作为示范，解释我们如何对于数据进行预处理：

组号	编号	厚度 mm	孔隙率 (%)	压缩回弹性 率 (%)	过滤阻 力 Pa	过滤效率 (%)	透气性 mm/s	插层率 (%)
1	1#	1.715	93.52	77.84	8.13	4.967	777.1	
	2#	2.81	96.28	83.2	7.533	19.967	1019.67	36.44

表二 原数据样例

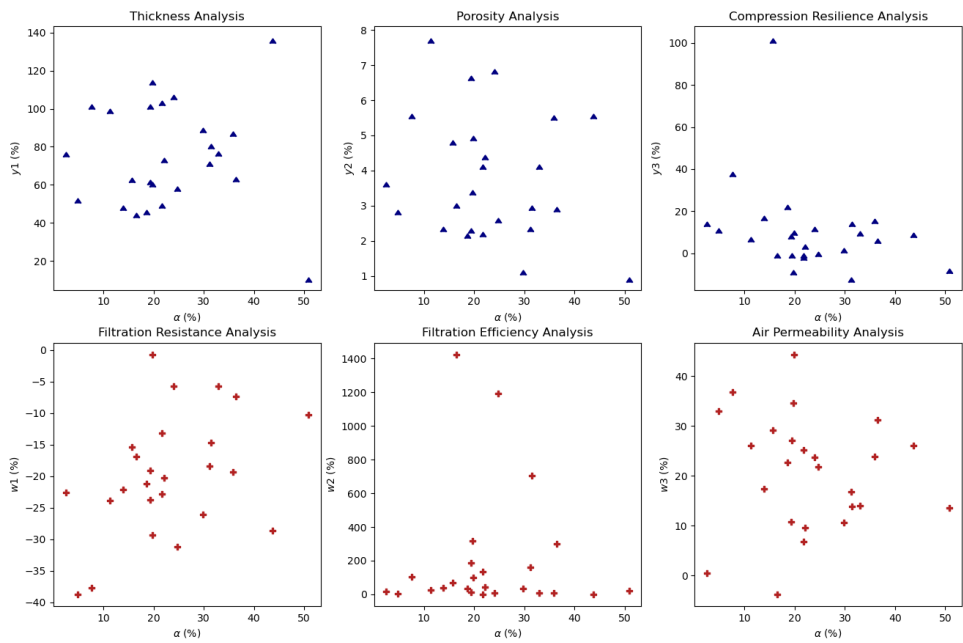
对于单位是百分比（%）的结构变量或工艺参数而言，我们直接用 2#组别的数据减去 1#组别的数据，得到插层后相应结构变量或工艺参数提高的百分比；对于单位不是百分比（%）的结构变量或工艺参数，我们用 2#组别的数据减去 1#组别的数据后，再除以 1#组别的数据，得到插层后相应结构变量或工艺参数提高的百分比，处理结果如下表所示：

组号	编号	厚度 mm	孔隙率 (%)	压缩回弹性 率 (%)	过滤阻力 Pa	过滤效率 (%)	透气性 mm/s	插层率 (%)
1	1#	1.715	93.52	77.84	8.13	4.967	777.1	
	2#	2.81	96.28	83.2	7.533	19.967	1019.67	36.44
	提高的 百分比	63.8484	2.76	5.36	-7.343173432	15	31.214773	

表三 预处理后数据样例

4.1.2 问题一第一小问模型求解及结论

以提高的百分比作为因变量，插层率提高的百分比作为自变量，通过统计绘制 6 个百分比图，论证正负相关性，解决变化规律问题，作图结果如下图所示。



图一 描述结构变量或工艺参数与插层率增加关系的散点图

由上述 6 个图可以定性分析 6 个结构变量或工艺参数与插层率增加的正负相关性。其中，除了过滤阻力与插层率显示的是负相关，其余五个结构变量或工艺参数（厚度，孔隙率，压缩回弹性率，过滤效率，透气性）都显示为正相关。

4.1.3 第二小问的模型建立与求解

Step 1: 确定分析数列

(1) 母序列(又称参考序列、母指标)，能反映系统行为特征的数据序列，类似于因变量 Y，此处记为 x_0 。

(2) 子序列(又称比较序列、子指标)

影响系统行为的因素组成的数据序列，类似于因变量 X，此处依次记为 x_1, x_2 等等。在本例中，我们依次将各个结构变量或工艺参数设置为母序列，其余结构变量或工艺参数和插层率变化设为子序列，共进行 6 组灰色关联度分析。

Step 2: 计算子序列中各个指标与母序列的关联系数

对母序列中的数据和子序列中的数据一次进行以下操作：
先通过计算式 $|x_0 - x_i|$ 得到母序列与子序列相对应组的差值，之后在每一组差值中定义：

$$a = \min_i \min_k |x_0(k) - x_i(k)|, b = \max_i \max_k |x_0(k) - x_i(k)|$$

其中 a 代表了两级最小差，b 代表了两级最大差。之后定义：

$$\gamma(x_0(k), x_i(k)) = \frac{a + \rho * b}{|x_0(k) - x_i(k)| + \rho * b}$$

其中ρ是分辨系数，一般取 0.5.

Step 3: 计算子序列中各个指标与母序列的灰色关联度

对每个指标的关联系数分别求平均值，得到灰色关联度如下列表所示：

	厚度 mm	孔隙率 (%)	压缩回弹性率 (%)	过滤阻力 Pa	过滤效率 (%)	透气性 mm/s
与插层率的灰色关联度	0.77510483	0.77510483	0.77510483	0.77510483	0.77510483	0.77510483

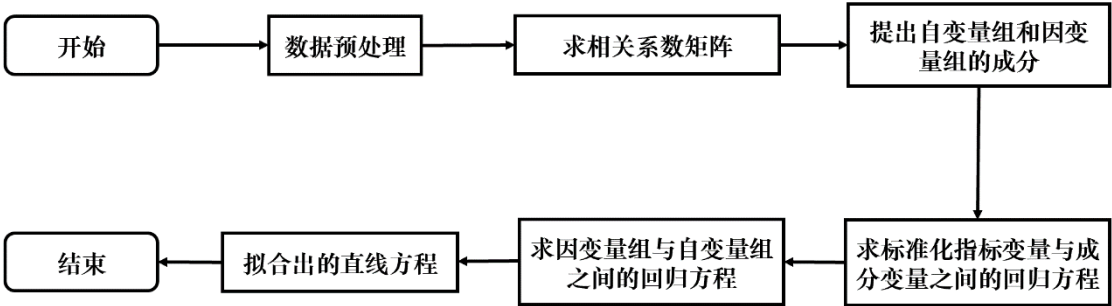
表四 各个指标与插层率的灰色关联度

以上灰色关联度均体现出插层率对于各个结构变量或工艺参数都存在较大影响。

4.2 问题二的分析与求解

4.2.1 方法选择的合理性

通过问题一中灰色关联度分析，各个结构变量或工艺参数之间存在较大关联度。而多元线性回归的方法则将因变量当作互相之间完全没有关联的变量进行处理和拟合，并不适用于本题情景。故本题选用偏最小二乘线性回归（PLS）的方法来替代多元线性回归的方法，在原有基础上工艺参数和结构变量之间的关系，并对表中 8 个工艺参数条件下的结构变量进行预测。



4.2.2 构造偏最小二乘模型

自变量的观察数据矩阵为 $A_1 = (a_{ij})_{75 \times 2}$ ，因变量的观测数据矩阵极为 $B_1 = (b_{ij})_{75 \times 3}$ 。

Step 1: 数据标准化

将各指标值转化为标准化的指标值，有：

$$\widetilde{a}_{ij} = \frac{a_{ij} - \mu_j^{(1)}}{s_j^{(1)}}, i = 1, 2, \dots, 75, j = 1, 2$$

其中：

$$\mu_j^{(1)} = \frac{1}{75} \sum_{i=1}^{75} a_{ij}; s_j^{(1)} = \sqrt{\frac{1}{75-1} \sum_{i=1}^{75} (a_{ij} - \mu_j^{(1)})^2} \quad j = 1, 2$$

即 $\mu_j^{(1)}, s_j^{(1)}$ 为第 j 个自变量 x_j 的样本均值和样本标准差。

因此，我们易得：

$$x_j = \frac{x_j - \mu_j^{(1)}}{s_j^{(1)}}, j = 1, 2$$

此为标准化的工艺参数指标。同样的道理，对于因变量我们采取同样的标准化操作，有

$$b_{ij} = \frac{b_{ij} - \mu_j^{(2)}}{s_j^{(2)}}, i = 1, 2, \dots, 75, j = 1, 2, 3$$

其中：

$$\mu_j^{(2)} = \frac{1}{75} \sum_{i=1}^{75} b_{ij}; s_j^{(2)} = \sqrt{\frac{1}{75-1} \sum_{i=1}^{75} (b_{ij} - \mu_j^{(2)})^2} \quad j = 1, 2, 3$$

即 $\mu_j^{(2)}, s_j^{(2)}$ 为第 j 个自变量 y_j 的样本均值和样本标准差。

$$\tilde{y}_j = \frac{y_j - \mu_j^{(2)}}{s_j^{(2)}}, j = 1, 2, 3$$

此为标准化的结构变量指标。

Step 2:求相关系数矩阵

	热风速度	厚度	孔隙率	压缩回弹性
接受距离	0	0.8169	0.73	-0.4026
热风速度	1	0.5551	0.5219	-0.3203
厚度	0.5551	1	0.8969	-0.527
孔隙率	0.5219	0.8969	1	-0.3236
压缩回弹性	-0.3203	-0.527	-0.3236	1

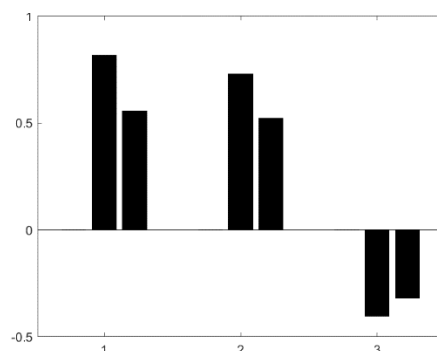
表五 相关系数矩阵

从相关系数矩阵中可以看出，接受距离与厚度和孔隙率成正比，与压缩回弹性成反比；热风速度与厚度和孔隙率成正比，与压缩回弹性成反比。

Step 3: 提出自变量组和因变量组的成分

分别提取出自变量组和因变量组的成分。其中第一个成分解释自变量的比率为 50%，前 2 个成分解释自变量的的比率为 100%，所以我们取两对成分。

$$\begin{cases} u_1 = -0.0949\tilde{x}_1 - 0.0672\tilde{x}_2 \\ v_1 = -8.4949\tilde{y}_1 - 7.7199\tilde{y}_2 + 4.4187\tilde{y}_3 \\ \begin{cases} u_2 = -0.0672\tilde{x}_1 + 0.0949\tilde{x}_2 \\ v_2 = -0.1620\tilde{y}_1 + 0.0364\tilde{y}_2 - 0.2748\tilde{y}_3 \end{cases} \end{cases}$$



图二 偏最小二乘回归系数直方图

Step 4: 求标准化指标变量与成分变量之间的回归方程

$$\begin{cases} \tilde{x}_1 = -7.0214u_1 - 4.9698u_2 \\ \tilde{x}_2 = -4.9698u_1 + 7.0214u_2 \\ \tilde{y}_1 = -8.4949u_1 - 0.1620u_2 \\ \tilde{y}_2 = -7.7199u_1 + 0.0364u_2 \\ \tilde{y}_3 = +4.4187u_1 - 0.2478u_2 \end{cases}$$

Step 5: 求因变量组与自变量组之间的回归方程

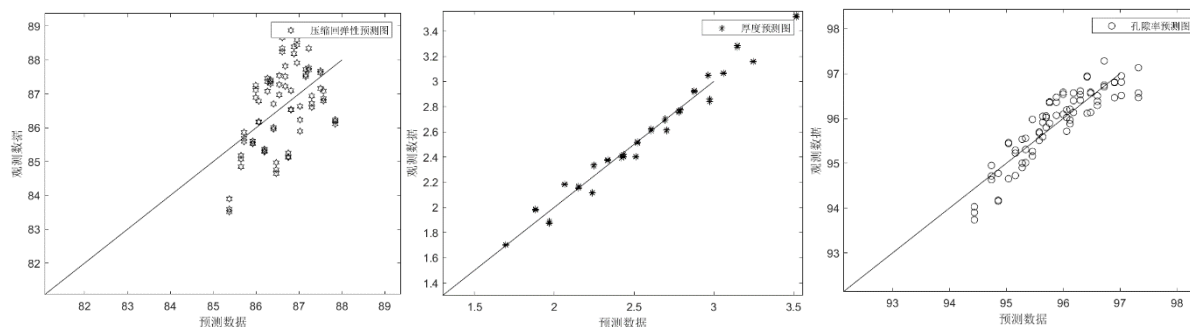
把步骤 3 中的 u_1 代入步骤 4 中的 \tilde{y}_1 的回归方程，得到标准化指标变量之间的回归方程：

$$\begin{cases} \tilde{y}_1 = 0.8169\tilde{x}_1 + 0.5551\tilde{x}_2 \\ \tilde{y}_2 = 0.7300\tilde{x}_1 + 0.5291\tilde{x}_2 \\ \tilde{y}_3 = -0.4026\tilde{x}_1 - 0.2478\tilde{x}_2 \end{cases}$$

将标准化变量还原为原始变量得：

$$\begin{cases} y_1 = -0.8601 + 0.0542x_1 + 0.0018x_2 \\ y_2 = 90.3490 + 0.0841x_1 + 0.0030x_2 \\ y_3 = 91.4094 - 0.0688x_1 - 0.0027x_2 \end{cases}$$

拟合出的直线方程如下图所示：



图三 结构变量预测图

4.2.3 问题求解

Part1：模型解释

我们绘制了回归系数的直方图，方便我们直观迅速地观察自变量再解释因变量时的边际作用。从回归系数的直方图中可以观察到接收距离在解释因变量时发挥着更大的作用，在所以 3 个因变量上回归系数的绝对值都是最大的。并且可以看出两个自变量对厚度和孔隙率的解释能力很充分，对于压缩回弹性率则稍弱一些。

除此之外，我们还绘制了预测图来检验模型的精度。在预测图上如果所有的点在对角线附近分布则说明模型拟合得很好。在 3 个预测中可以观察到点都分布在对角线附近，其中厚度和孔隙率的效果非常好，压缩回弹性率则稍弱一些。这样的结果和回归系数直方图呈现的自变量对因变量解释能力的强弱是一致的。

Part2：结构变量数据预测

我们根据模型预测出的结构变量的数据如下表所示。

接受距离	厚度	孔隙率	压缩回弹性
38	2. 7295	96. 0948	86. 5122
33	2. 6385	95. 9743	86. 574
28	2. 7275	96. 1538	86. 3782
23	2. 6365	96. 0333	86. 4523
38	3. 4495	97. 2948	85. 4242
33	3. 1785	96. 8743	85. 7646
28	2. 3675	95. 5538	86. 9181
23	1. 9165	94. 8333	87. 5327

表六 问题二预测结论

5.3 问题三的建模与求解

5.3.1 问题分析与模型的选用

问题三要求我们分析工艺参数与产品性能之间的关系，以及结构变量之间和产品性能之间的关系。类似于第二小问，通过问题一中灰色关联度分析，各个工艺参数之间存在较大关联度。而多元线性回归的方法则将因变量当作互相之间完全没有关联的变量进行处理和拟合，并不适用于本题情景。对于本题，我们仍然采用偏最小二乘的方法进行研究。

5.3.2 模型的构建与求解

Step 1：数据标准化

同 5.2.2 中的 Step1.

Step 2: 求相关系数矩阵

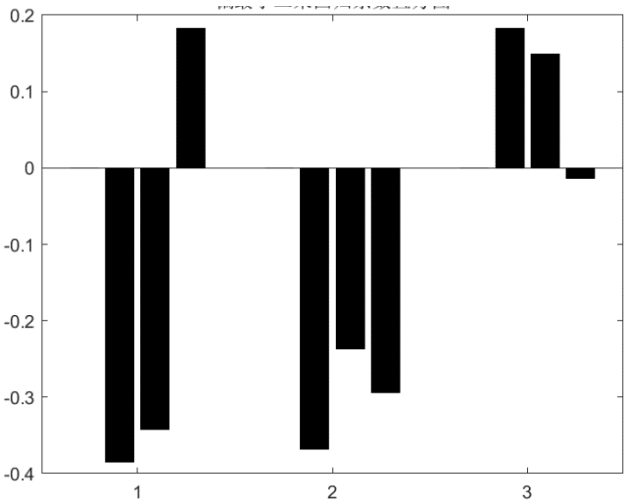
	孔隙率	压缩回弹性	过滤阻力	过滤效率	透气性
厚度	0.8969	-0.527	-0.7952	-0.4826	0.3537
孔隙率	1	-0.3236	-0.7365	-0.4007	0.2787
压缩回弹性	-0.3236	1	0.4975	-0.0101	-0.1652
过滤阻力	-0.7365	0.4975	1	0.374	-0.4104
过滤效率	-0.4007	-0.0101	0.374	1	-0.789
透气性	0.2787	-0.1652	-0.4104	-0.789	1

表七 相关系数矩阵

Step 3: 提出自变量组和因变量组的成分

分别提取出自变量组和因变量组的成分。其中第一个成分解释自变量的比率为 73.12%，前 2 个成分解释自变量的的比率为 97.15%，所以我们取两对成分。

$$\begin{cases} u_1 = -0.0555\widetilde{y}_1 - 0.0494\widetilde{y}_2 + 0.0264\widetilde{y}_3 \\ v_1 = 6.9275\widetilde{w}_1 + 3.4264\widetilde{w}_2 - 2.7932\widetilde{w}_3 \\ u_2 = -0.0605\widetilde{y}_1 - 0.0231\widetilde{y}_2 - 0.1308\widetilde{y}_3 \\ v_2 = -1.0451\widetilde{w}_1 + 2.5835\widetilde{w}_2 - 0.7436\widetilde{w}_3 \end{cases}$$



图四 偏最小二乘回归系数直方图

Step 4: 求标准化指标变量与成分变量之间的回归方程

求得自变量组和因变量组与 u_1, u_2 之间的回归方程分别为：

$$\begin{cases} \tilde{y}_1 = -8.4122u_1 - 0.9085u_2 \\ \tilde{y}_2 = -7.9687u_1 - 2.5930u_2 \\ \tilde{y}_3 = +5.2979u_1 - 6.7670u_2 \\ \tilde{w}_1 = +6.9275u_1 + 0.0032u_2 \\ \tilde{w}_2 = +3.4264u_1 - 2.9429u_2 \\ \tilde{w}_3 = -2.7932u_1 - 0.4601u_2 \end{cases}$$

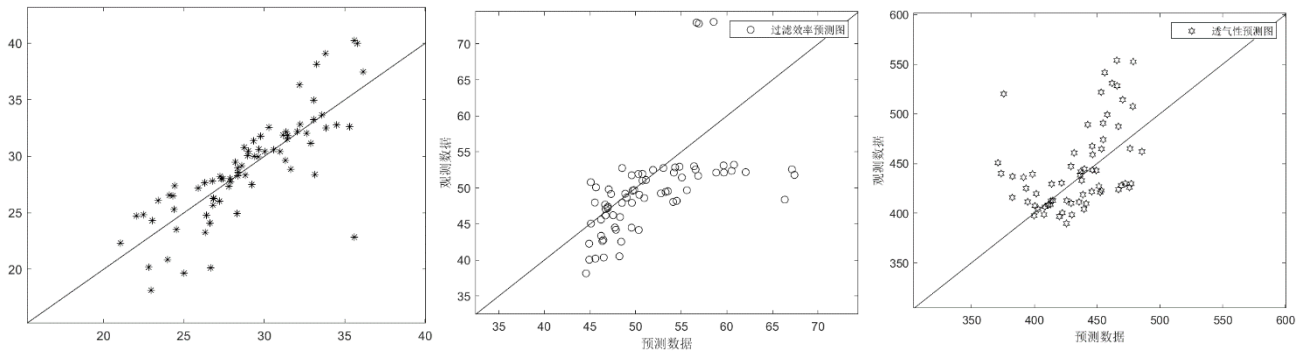
Step 5: 求因变量组与自变量组之间的回归方程

把步骤 3 中的 u_i 代入步骤 4 中的 \tilde{w}_i 的回归方程，得到标准化指标变量之间的回归方程：

$$\begin{cases} \tilde{w}_1 = -0.3846\tilde{y}_1 - 0.3422\tilde{y}_2 + 0.1823\tilde{y}_3 \\ \tilde{w}_2 = -0.3680\tilde{y}_1 - 0.2373\tilde{y}_2 - 0.2945\tilde{y}_3 \\ \tilde{w}_3 = +0.1828\tilde{y}_1 + 0.1486\tilde{y}_2 - 0.0135\tilde{y}_3 \end{cases}$$

将标准化变量还原为原始变量得：

$$\begin{cases} w_1 = 163.0440 - 3.7479y_1 - 1.9200y_2 + 0.6894y_3 \\ w_2 = 598.6084 - 8.3956y_1 - 3.1170y_2 - 2.6076y_3 \\ w_3 = -1052.6902 + 33.0318y_1 + 15.4584y_2 - 0.9455y_3 \end{cases}$$



图五 绘制预测图检验模型精度

5.3.3 问题求解

Part 1: 模型解释

与问题二中的分析原则一致。我们绘制了回归系数的直方图，方便我们直观迅速地观察自变量再解释因变量时的边际作用。从回归系数的直方图中可以观察到厚度和孔隙率在解释因变量时发挥着较大的作用，并且结厚度和孔隙率在对于不同因变量的解释能力是趋同的(两个方面:回归系数的正负以及绝对值的大小)，这一点也可以从它们的相关系数为 0.8969 看出来。另外，比较特别的一点是，压缩回弹性率对于不同的因变量呈现非常不同的解释能力。对于过滤效率，压缩回弹性率和厚度、孔隙率一样展现出高解释能力，但是对于透气性的解释能力却微乎其微。这种复杂的，局部的相关性在偏最小二乘回归模型中得到有限的展现，体现了我们模型的优越性。

同理，我们还绘制了预测图来检验模型的精度。在预测图上如果所有的点在对角线附近分布则说明模型拟合得很好。在 3 个预测图中可以观察到点基本分布在对角线附近，对比结构变量和工艺参数的模型，虽然模型表现有所下降但是整体来看仍然有着较高的

精度。对于模型精确度略微下降的分析如下：

我们从附件中的数据可以观察到接受距离为 20cm，热风速度从 900r/min 变为 1000r/min 时，过滤效率发生了跳变，从 52.366%变为 72.801%。接受距离为 20cm，热风速度从 1000r/min 变为 1100r/min 时，透气性从 408.609mm/s 跳变为 235.229mm/s。这两个数值的跳变我们合理猜测是因为条件达到了临界值导致发生了相应的化学反应或者物理反应。由于我们的模型为统计模型，所以没有办法去学习这些跳变的规律，只能尽可能去拟合跳变前后的数值差异。因此，这一点上使得模型的精确度略微下降。

当然在整体上模型是精确的。而且，虽然没有对于跳变的因变量的值进行精确预测，但是我们的模型仍然可以拟合得到相应的趋势。换句话说，在对跳变的因变量值预测欠缺精度的情况下，我们仍然可以精确的预测出因变量(产品性能)取最优时自变量(结构变量)为多少。这也为我们第四问的多目标优化问题设立了模型基础。

Part 2：结构变量之间的关系

根据相关系数矩阵可以看出，厚度和孔隙率的相关系数是 0.8969，有很强的正相关性。厚度和压缩回弹性的相关系数是-0.5270，呈负相关性。孔隙率和压缩回弹性的相关系数为-0.3236，呈负相关性，但相关性较弱。

Part 3：产品参数之间的关系

同理，根据步骤 2 中绘制的相关系数矩阵。过滤阻力和过滤效率相关系数为 0.3740，呈正相关性，但相关性较弱。过滤阻力和透气性的相关系数为-0.4104，呈负相关性，但相关性较弱。过滤效率和透气性的相关系数为-0.7890，有很强的负相关性。

Part 4：参数选择

我们已经建立了关于产品性能和结构变量之间的模型，针对于问题二中的数据，我们在预测出的结构变量的基础上可以计算相应的产品性能从而挑选出最优的工艺参数。计算的结果如下表所示：

接受距离	过滤阻力	过滤效率	透气性
38	27.9452	50.6077	441.1562
33	28.5686	51.5543	436.2176
28	27.7552	50.7587	442.1175
23	28.3787	51.7053	437.1789
38	22.1981	43.6386	484.5103
33	24.2583	46.3276	468.7332
28	30.6288	54.2432	420.4405

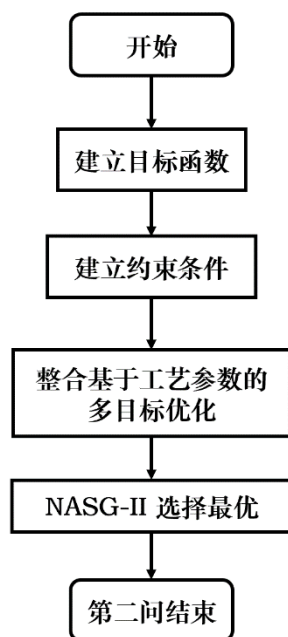
表八 接受距离与产品性能间的关系

由表格结果可知，过滤效率最高时，接收距离为 23cm，热风速度为 850r/min。

5.4 问题四的求解

5.4.1 问题分析

为了满足产品生产需要兼顾各方面的条件和要求，保证工艺参数的限制和应用的要求，同时追求较高的过滤效率和较小的过滤阻力的目标。我们建立了基于过滤效率和过滤阻力的多目标优化模型，通过调整工艺参数，在保证约束条件的前提下，使得过滤效率尽量的高的同时力求过滤阻力尽量的小，得到最优的工艺参数，进而优化产品性能。



5.4.2 模型的建立与求解

Part 1: 建立目标函数

Step 1: 工艺参数与结构变量的模型

根据第二问的求解，我们易知结构变量厚度(y_1)、孔隙率(y_2)、压缩回弹性率(y_3)与工艺参数接收距离(x_1)、热风速度(x_2)之间的关系如下：

$$\begin{cases} y_1 = -0.8601 + 0.0542x_1 + 0.0018x_2 \\ y_2 = 90.3490 + 0.0841x_1 + 0.0030x_2 \\ y_3 = 91.4094 - 0.0688x_1 - 0.0027x_2 \end{cases}$$

Step 2: 结构变量与产品性能的模型

根据第三问的求解，我们易知产品性能过滤阻力(w_1)、过滤效率(w_2)与结构变量厚度(y_1)、孔隙率(y_2)、压缩回弹性率(y_3)之间的关系如下：

$$\begin{cases} w_1 = 163.0440 - 3.7479y_1 - 1.9200y_2 + 0.6894y_3 \\ w_2 = 598.6084 - 8.3956y_1 - 3.1170y_2 - 2.6076y_3 \end{cases}$$

Step 3: 产品性能与工艺参数的模型

因为工艺参数决定结构变量，而由结构变量决定最终产品性能，所以，在本问题中，

我们统一工艺参数与结构变量的模型和结构变量与产品性能的模型，将中间的结构变量的模型由工艺参数的模型替代，最终，我们得到产品性能过滤阻力(w_1)、过滤效率(w_2)与工艺参数接收距离(x_1)、热风速度(x_2)之间的关系如下：

$$\begin{cases} w_1 = 55.8152 - 0.4127 x_1 - 0.0141 x_2, \\ w_2 = 85.8521 - 0.5382 x_1 - 0.0174 x_2, \end{cases}$$

Step 4: 目标函数确定

产品的目的是需要同时追求过滤效率高和过滤阻力小的目标。所以，我们最终的目标函数为：

$$\begin{cases} \min w_1 = 55.8152 - 0.412 x_1 - 0.0141 x_2, \\ \max w_2 = 85.8521 - 0.538 x_1 - 0.0174 x_2 \end{cases}$$

Part 2: 建立约束条件

首先，我们根据题意对于工艺参数和结构变量的常规范范围构建约束。在约束限制中，我们同目标函数一样，统一将结构变量以工艺参数进行整合表示：

约束条件①：接收距离不大可能大于 100cm

$$0 \leq x_1 \leq 100$$

约束条件②：热空气速度也不大可能大于 2000r/min

$$0 \leq x_2 \leq 2000$$

约束条件③：厚度尽量不要超过 3mm

$$0 \leq y_1 \leq 3$$

$$\Rightarrow 0 \leq -0.8601 + 0.0542x_1 + 0.0018x_2 \leq 3$$

约束条件④：压缩回弹性率尽量不要低于 85%

$$85 \leq y_3 \leq 100$$

$$\Rightarrow 85 \leq 91.4094 - 0.0688x_1 - 0.0027x_2 \leq 100$$

约束条件⑤：对于结构变量厚度，我们应保证其范围位于 0-100 的范围内

$$0 \leq y_2 \leq 100$$

$$\Rightarrow 0 \leq y_2 = 90.3490 + 0.0841x_1 + 0.0030x_2 \leq 100$$

Part 3: 整合基于工艺参数的多目标优化

通过整合上述目标函数与约束条件，我们得到了基于工艺参数的多目标优化模型：

$$\begin{cases} \min w_1 = 55.8152 - 0.412 x_1 - 0.0141 x_2 \\ \max w_2 = 85.8521 - 0.538 x_1 - 0.0174 x_2 \\ 0 \leq x_1 \leq 100 \\ 0 \leq x_2 \leq 2000 \\ 0 \leq -0.8601 + 0.0542x_1 + 0.0018x_2 \leq 3 \\ 85 \leq 91.4094 - 0.0688x_1 - 0.0027x_2 \leq 100 \\ 0 \leq y_2 = 90.3490 + 0.0841x_1 + 0.0030x_2 \leq 100 \end{cases}$$

Part 4: NASG-II 绘制 Pareto 前沿

在多目标优化问题的求解中，在本实验中，我们采用 NASG-II 来绘制 Pareto 前沿。NSGA-II 算法，即带有精英保留策略的快速非支配多目标优化算法，是一种基于 Pareto 最优解的多目标优化算法。NSGA-II 算法的基本思想为：首先，随机产生规模为 N 的初始种群，非支配排序后通过遗传算法的选择、交叉、变异三个基本操作得到第一代子代

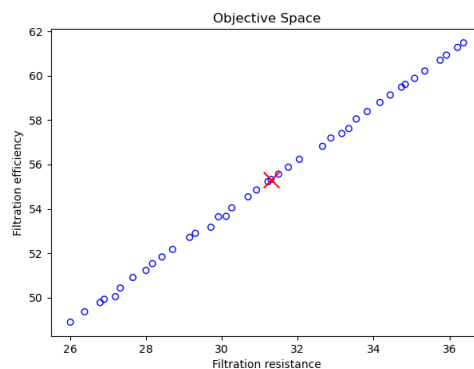
种群；其次，从第二代开始，将父代种群与子代种群合并，进行快速非支配排序，同时对每个非支配层中的个体进行拥挤度计算，根据非支配关系以及个体的拥挤度选取合适的个体组成新的父代种群；最后，通过遗传算法的基本操作产生新的子代种群：依此类推，直到满足程序结束的条件。算法的伪代码如下所示：

Algorithm 1 NASG-II 算法

Require: 确定种群大小 n ，交叉概率 t ，迭代次数 g

- 1: 随机产生 n 个个体，它们整体视为种群 P
 - 2: **for** $i = 1$ to g **do**:
 - 3: $P = \emptyset$
 - 4: **for** $j = 1$ in n **do**:
 - 5: 产生一个 $[0, 1]$ 的随机数 a
 - 6: **if** $a < t$ **then**
 - 7: 从 P 中随机选出两个个体作为父母，交叉产生一个新的个体并放入 P' 中
 - 8: **else** 从 P 中随机选出一个个体，变异产生一个新的个体并放入 P' 中
 - 9: **end if**
 - 10: **end for**
 - 11: 利用非支配排序和拥挤距离，从 $P \cup P'$ 中选出 n 个个体，代替 P
 - 12: **end for**
 - 13: 输出最终种群 P 中的非支配个体
-

在整合基于工艺参数的多目标优化模型的优化任务中，我们输出的 Pareto 前沿如下所示：



图六 整合基于工艺参数的多目标优化模型的 pareto 前沿，红色点为我们选择的最优点

Part 5; Pareto 前沿最优点选取

在本实验中，我们认为过滤阻力和过滤效率同样重要，所以，在本实验中，我们选取 pareto 前沿的中间点作为最优点，使得过滤效率尽量的高的同时，过滤阻力尽量的小。其对应的工艺参数如下：

工艺参数	接受距离(cm)	热风速度(r/min)
<hr/>		

结果	20.4016	1168.3642
----	---------	-----------

表九 帕累托前沿最优点

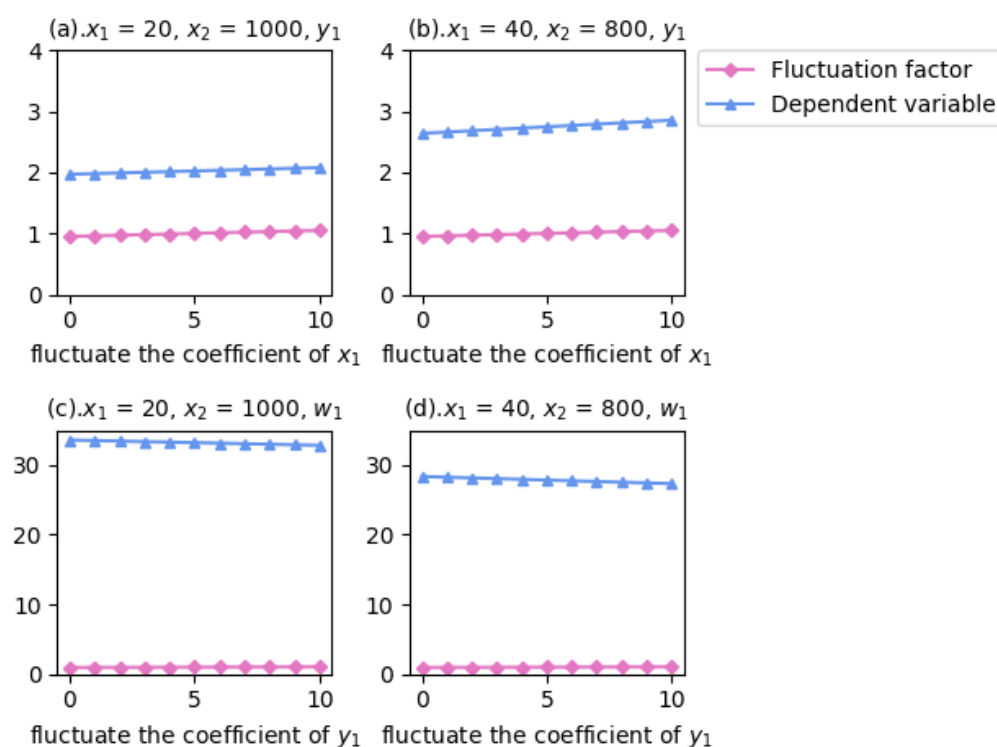
最终，我们预测最优情况下的产品性能如下：

产品性能	过滤阻力(Pa)	过滤效率(%)
结果	31.2172	55.2289

表十 最优产品性能

6 敏感性分析

根据问题二和问题三的分析，我可以看出：对于结构变量-工艺参数的模型，自变量接收距离对因变量厚度有着强烈的影响。对于产品性能-结构变量的模型，自变量厚度对因变量过滤阻力也有着强烈的影响。因此我们分别对于厚度和过滤阻力的回归方程进行敏感度分析。具体来讲，在厚度的回归方程中，我们使接受距离的回归系数上下波动5%。在过滤阻力的回归方程中，我们使厚度的回归系数上下波动5%。然后作出敏感度的分析如下图：



图七 敏感度的分析

为了使得我们的说明更具广泛性，我们设定 $x_1 = 20, x_2 = 1000$ ，以及 $x_1 = 40, x_2 = 800$ 两组相差较大的初始工艺参数。

蓝色的线是因变量的波动曲线，紫色的线是回归系数的波动曲线（波动范围为

0.95~1.05)。以左上角的图为例，我们的研究对象为 y_1 ，即厚度，则蓝色的曲线代表着厚度随着 x_1 的系数波动而波动。

研究的因变量的波动范围	
$x_1 = 20, x_2 = 1000, x_1$ 系数波动范围 10%, 研究 y_1	2.6394~2.8563, +8.21%
$x_1 = 40, x_2 = 800, x_1$ 系数波动范围 10%, 研究 y_1	1.9697~2.0781, +5.50%
$x_1 = 20, x_2 = 1000, x_1$ 系数波动范围 10%, 研究 w_1	28.3544~27.3245, -3.63%
$x_1 = 40, x_2 = 800, x_1$ 系数波动范围 10%, 研究 w_1	33.5860~32.8274, -2.26%

表十一 研究的因变量的波动范围

结合上文图表，可以看出因变量的波动的范围都小于回归系数波动的范围，比较稳定，证明该模型很好的稳定性。

7 结论

【第一问的结论】

- (1) 除了过滤阻力与插层率显示的是负相关，其余五个结构变量或工艺参数（厚度，孔隙率，压缩回弹性率，过滤效率，透气性）都显示为正相关。
- (2) 由灰色关联度分析体现出插层率对于各个结构变量或工艺参数都存在较大影响。

【第二问的结论】

接受距离	厚度	孔隙率	压缩回弹性
38	2.7295	96.0948	86.5122
33	2.6385	95.9743	86.574
28	2.7275	96.1538	86.3782
23	2.6365	96.0333	86.4523
38	3.4495	97.2948	85.4242
33	3.1785	96.8743	85.7646
28	2.3675	95.5538	86.9181
23	1.9165	94.8333	87.5327

【第三问的结论】

- (1) 根据相关系数矩阵可以看出，厚度和孔隙率的相关系数是 0.8969，有很强的正相关性。厚度和压缩回弹性的相关系数是-0.5270，呈负相关性。孔隙率和压缩回弹性的相关系数为-0.3236，呈负相关性，但相关性较弱。

- (2) 同理，根据步骤 2 中绘制的相关系数矩阵。过滤阻力和过滤效率相关系数为 0.3740，呈正相关性，但相关性较弱。过滤阻力和透气性的相关系数为-0.4104，呈负相关性，但相关性较弱。过滤效率和透气性的相关系数为-0.7890，有很强的负相关性。
- (3) 过滤效率最高时，接收距离为 23cm，热风速度为 850r/min。

【第四问的结论】

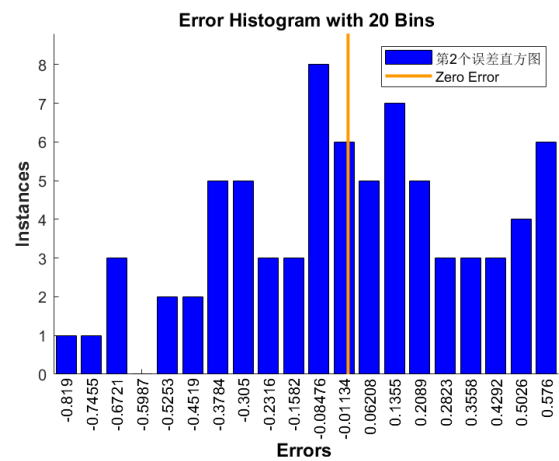
工艺参数中，接受距离为 20.4016cm、热风速度为 1168.3642r/min 时，产品性能中过滤阻力为 31.2172pa，过滤效率为 55.2289%。

参考文献

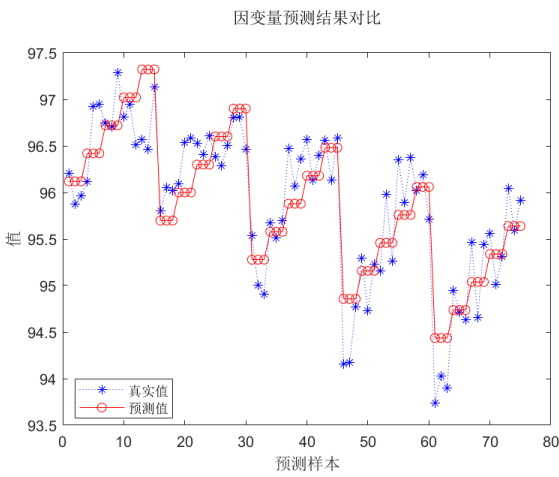
- [1] 练军, 陆忠, 熔喷非织造布过滤性能的研究, 产业用纺织品, 20(9):13-5, 2002
- [2] 曹明霞, 灰色关联分析模型及其应用的研究, 南京: 南京航空航天大学, 2007
- [3] Hulland J, Use of partial least squares (PLS) in strategic management research: A review of four recent studies, Strategic management journal, Feb;20(2):195-204, 1999
- [4] 蒋红卫, 夏结来, 偏最小二乘回归及其应用, 第四军医大学学报, 24(3):280-3, 2003
- [5] 严丽坤, 相关系数与偏相关系数在相关分析中的应用, 云南财贸学院学报, 19(3):78-80, 2003
- [6] 公茂果, 焦李成, 杨咚咚, 马文萍, 进化多目标优化算法研究, 软件学报, 20(2):271-89, 2009
- [7] Deb K, Pratap A, Agarwal S, Meyarivan TA, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE transactions on evolutionary computation, Aug 7;6(2):182-97.2002
- [8] Blank J, Deb K, Pymoo: Multi-objective optimization in python, IEEE Access, 8:89497-509, 2020

附录

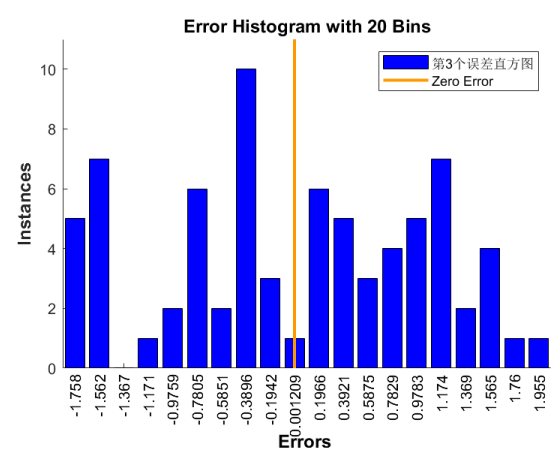
A. 图像补充



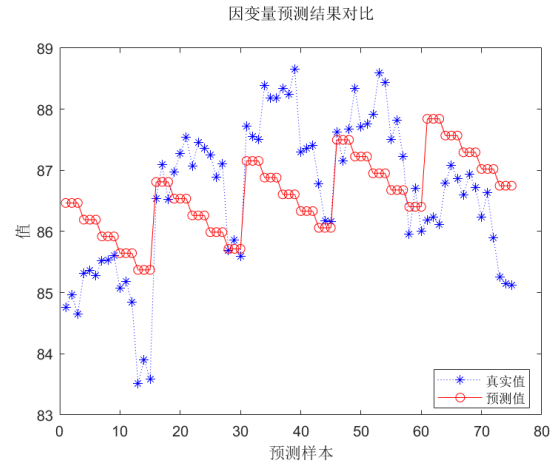
图一 问题 2 的第 2 个因变量的误差直方图



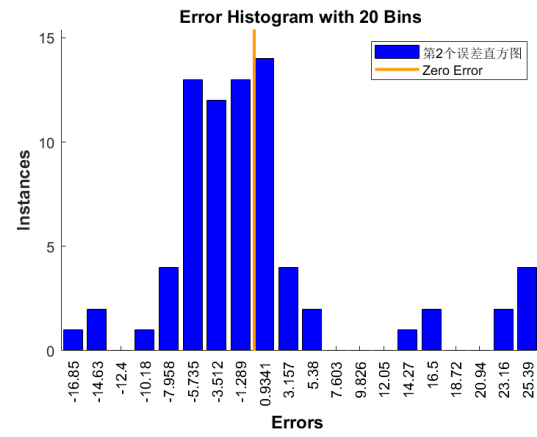
图二 问题 2 的第 2 个因变量的回归图



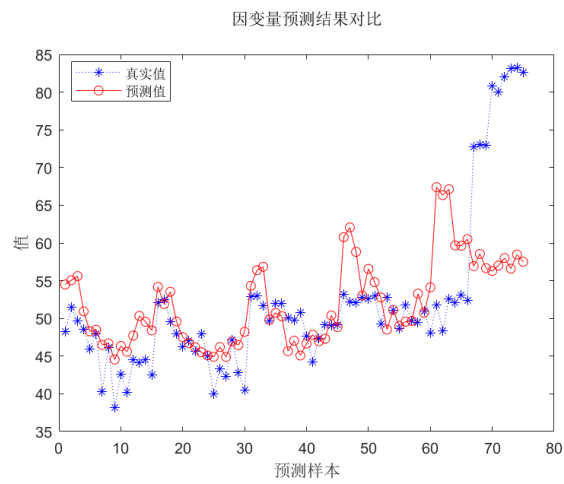
图三 问题 2 的第 3 个因变量的误差直方图



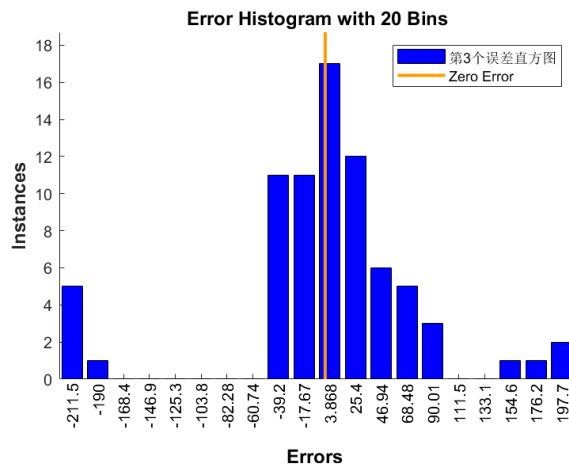
图四 问题 2 的第 3 个因变量的回归图



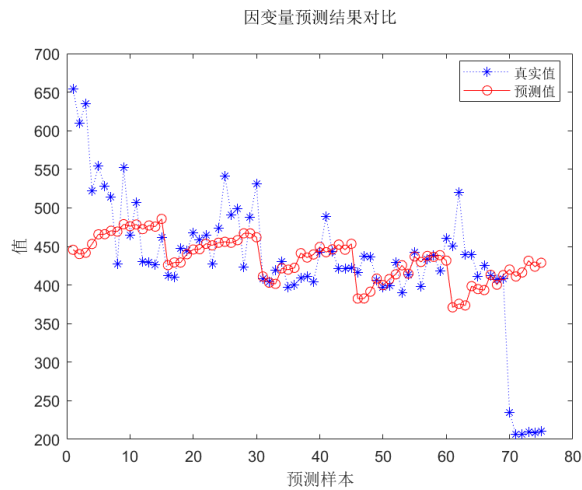
图五 问题 3 的第 2 个因变量的误差直方图



图六 问题 3 的第 2 个因变量的回归图



图七 问题3的第3个因变量的误差直方图



图八 问题3的第3个因变量的回归图

B. 相关源代码

1. 第一问中使用统计学算法研究插层后结构变量、产品性能的变化规律代码

import numpy as np
import math
import matplotlib.pyplot as plt
from scipy.stats import pearsonr
x_data = []
y_data = []
temp = []
f = open(r'data.txt', 'r') #以读方式打开文件
for line in f.readlines(): #依次读取每行
l = line.split('\t')
l[-1] = l[-1].replace('\n', '')

temp.append(l)
fig = plt.figure(figsize=(16, 13))
ax = fig.subplots(2, 3) # 2*2
name = ['Thickness Analysis', 'Porosity Analysis', 'Compression Resilience Analysis', 'Filtration Resistance Analysis', 'Filtration Efficiency Analysis', 'Air Permeability Analysis',]
y_label = [r"\$y_1\$", r"\$y_2\$", r"\$y_3\$", r"\$w_1\$", r"\$w_2\$", r"\$w_3\$",]
参数调整
for num in range(1, 7):
for i in range(0, 25):
x = (eval(temp[i * 2 + 1][-1]) - eval(temp[i * 2][-1]))
y = (eval(temp[i * 2 + 1][num + 1]) - eval(temp[i * 2][num + 1])) / eval(temp[i * 2][num + 1]) * 100
x_data.append(x)
y_data.append(y)
通过切片获取横坐标 x1
x1 = x_data
通过切片获取纵坐标 R
y1 = y_data
ax[int((num - 1) / 3), (num - 1) % 3].set_title(name[num-1])
ax[int((num - 1) / 3), (num - 1) % 3].set_xlabel(r"\$\alpha\$" + ' (%)')
ax[int((num - 1) / 3), (num - 1) % 3].set_ylabel(y_label[num-1] + ' (%)')
if num < 4:
ax[int((num - 1) / 3), (num - 1) % 3].scatter(x1, y1, marker=6, color='navy')
else:
ax[int((num - 1) / 3), (num - 1) % 3].scatter(x1, y1, marker='P', color='firebrick')
x_data = []
y_data = []
plt.show()

2. 第一问中使用灰色关联分析算法分析影响代码

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import matplotlib as mpl
mpl.rcParams['font.sans-serif'] = ['SimHei']
mpl.rcParams['axes.unicode_minus'] = False

data = pd.read_excel('data.xlsx')
提取变量名
label_need = data.keys()[0:]
提取上面变量名下的数据
data1 = data[label_need].values
数据归一化
[m, n] = data1.shape
data2 = data1.astype('float')
data3 = data2
ymin = 0.002
ymax = 1
for j in range(0,n):
d_max = max(data2[:, j])
d_min = min(data2[:, j])
data3[:, j] = (ymax - ymin) * (data2[:, j] - d_min) / (d_max - d_min) + ymin
得到其他列和参考列相等的绝对值
for i in range(0, 9):
data3[:, i] = np.abs(data3[:, i] - data3[:, 7])
#得到绝对值矩阵的全局最大值和最小值
data4 = data3[:, 0:9]
d_max = np.max(data2)
d_min = np.min(data2)
a = 0.5 #定义分辨系数
计算灰色关联矩阵
data4 = (d_min + a * d_max) / (data4 + a * d_max)
xishu = np.mean(data4, axis=0)
print('灰色关联度结果分别为: ')
print(xishu)

3. 第二问中建立结构变量-工艺参数的偏最小二乘回归模型代码

clc,clear
ab0=load('data1.txt');
mu=mean(ab0), sig=std(ab0)
rr=corrcoef(ab0)
ab=zscore(ab0);
a=ab(:,[1:2]);b=ab(:,[3:end]);

```

[XL,YL,XS,YS,BETA,PCTVAR,MSE,stats] =plsregress(a,b)
contr=cumsum(PCTVAR,2)
xw=a\XS
yw=b\YS
ncomp=input('请根确定提出成分对的个数 ncomp=');
[XL2,YL2,XS2,YS2,BETA2,PCTVAR2,MSE2,stats2] =plsregress(a,b,ncomp)
n=size(a,2); m=size(b,2);
beta3(1,:)=mu(n+1:end)-mu(1:n)./sig(1:n)*BETA2([2:end],:).*sig(n+1:end);
beta3([2:n+1],:)=(1./sig(1:n))*sig(n+1:end).*BETA2([2:end],:)
yhat=repmat(beta3(1,:),[size(a,1),1])+ab0(:,[1:n])*beta3([2:end],:)
y0 = ab0(:,end-size(yhat,2)+1:end)
for i =1:size(b,2)
    yz = y0(:,i);
    yc = yhat(:,i);
    N = size(a,1);
    perf = mse(y0,yhat)
    figure;
    plotregression(yz,yc,['第',num2str(i),'个回归图'])
    e = yz-yc;
    figure;
    ploterrhist(e,['第',num2str(i),'个误差直方图'])
    R2 = (N*sum(yc.*yz)-sum(yc)*sum(yz))^2/((N*sum((yc).^2)-(sum(yc))^2)*(N*sum((yz).^2)-(sum(yz))^2));
    figure;
    plot(1:N,yz,'b:*',1:N,yc,'r-o')
    legend('真实值','预测值','location','best')
    xlabel('预测样本')
    ylabel('值')
    string = {'因变量预测结果对比';[]};
    title(string)
end
ymax=max([yhat;ab0(:,n+1:end)]); %求预测值和观测值的最大值
bar(BETA2,'k') %画直方图
title("偏最小二乘回归系数直方图")
plot(yhat(:,1),ab0(:,n+1),'*',[0:ymax(1)],[0:ymax(1)],'Color','k')
legend('单杠成绩预测图'), xlabel('预测数据'), ylabel('观测数据')
%下面画 y1,y2,y3 的预测图，并画直线 y=x
figure, subplot(2,2,1)
plot(yhat(:,1),ab0(:,n+1),'*',[0:ymax(1)],[0:ymax(1)],'Color','k')
legend('厚度预测图'), xlabel('预测数据'), ylabel('观测数据')
subplot(2,2,2)
plot(yhat(:,2),ab0(:,n+2),'O',[0:ymax(2)],[0:ymax(2)],'Color','k')
legend('孔隙率预测图'), xlabel('预测数据'), ylabel('观测数据')

```


subplot(2,2,3)
plot(yhat(:,3),ab0(:,end),'H',[0:ymax(3)],[0:ymax(3)],'Color','k')
legend('压缩回弹性预测图'), xlabel('预测数据'), ylabel('观测数据')

4. 第三问中建立产品性能-结构变量的偏最小二乘回归模型代码

clc,clear
ab0=load('data2.txt');
mu=mean(ab0), sig=std(ab0)
rr=corrcoef(ab0)
ab=zscore(ab0);
a=ab(:,[1:3]);b=ab(:,[4:end]);
[XL,YL,XS,YS,BETA,PCTVAR,MSE,stats] =plsregress(a,b)
contr=cumsum(PCTVAR,2)
xw=a\XS
yw=b\YS
ncomp=input('请根据 PCTVAR 的值确定提出成分对的个数 ncomp=');
[XL2,YL2,XS2,YS2,BETA2,PCTVAR2,MSE2,stats2] =plsregress(a,b,ncomp)
n=size(a,2); m=size(b,2);
beta3(1,:)=mu(n+1:end)-mu(1:n)./sig(1:n)*BETA2([2:end],:).*sig(n+1:end);
beta3([2:n+1],:)=(1./sig(1:n))*sig(n+1:end).*BETA2([2:end],:)
bar(BETA2,'k')
title("偏最小二乘回归系数直方图")
yhat=repmat(beta3(1,:),[size(a,1),1])+ab0(:,[1:n])*beta3([2:end],:);
ymax=max([yhat;ab0(:,[n+1:end])]);
y0 = ab0(:,end-size(yhat,2)+1:end);
for i =1:size(b,2)
yz = y0(:,i);
yc = yhat(:,i);
N = size(a,1);
perf = mse(y0,yhat)
figure;
plotregression(yz,yc,['第',num2str(i),'个回归图'])
e = yz-yc;
figure;
ploterrhist(e,['第',num2str(i),'个误差直方图'])
$R^2 = (N*\sum(yz.*yc)-\sum(yz)*\sum(yc))^2/((N*\sum((yc).^2)-(\sum(yc))^2)*(N*\sum((yz).^2)-(\sum(yz))^2));$
figure;
plot(1:N,yz,'b:*',1:N,yc,'r-o')
legend('真实值','预测值','location','best')
xlabel('预测样本')
ylabel('值')
string = {'因变量预测结果对比';[]};

title(string)
end
plot(yhat(:,1),ab0(:,n+1),'*',[0:ymax(1)],[0:ymax(1)],'Color','k')
legend('过滤阻力预测图'), xlabel('预测数据'), ylabel('观测数据')
%下面画 y1,y2,y3 的预测图，并画直线 y=x
figure, subplot(2,2,1)
plot(yhat(:,1),ab0(:,n+1),'*',[0:ymax(1)],[0:ymax(1)],'Color','k')
legend('过滤阻力预测图',2), xlabel('预测数据'), ylabel('观测数据')
subplot(2,2,2)
plot(yhat(:,2),ab0(:,n+2),'O',[0:ymax(2)],[0:ymax(2)],'Color','k')
legend('过滤效率预测图'), xlabel('预测数据'), ylabel('观测数据')
subplot(2,2,3)
plot(yhat(:,3),ab0(:,end),'H',[0:ymax(3)],[0:ymax(3)],'Color','k')
legend('透气性预测图'), xlabel('预测数据'), ylabel('观测数据')

5. 第四问中使用 NSGA-II 多目标优化模型求解最优工艺参数代码

from pymoo.algorithms.moo.nsga2 import NSGA2
from pymoo.factory import get_sampling, get_crossover, get_mutation, get_problem
from pymoo.core.problem import ElementwiseProblem
from pymoo.optimize import minimize
import numpy as np
import matplotlib.pyplot as plt
import time
class MyProblem(ElementwiseProblem):
def __init__(self):
super().__init__(n_var=2, # 变量数
n_obj=2, # 目标数
n_constr=6, # 约束数
xl=np.array([20, 800]), # 变量下界
xu=np.array([100, 2000]), # 变量上界
)
def _evaluate(self, x, out, *args, **kwargs):
定义目标函数
f1 = (-0.412 * x[0] - 0.014 * x[1] + 55.815) # x1 放在 x 的第 0 列，x2 放在 x 的第一列
f2 = -1 * (-0.538 * x[0] - 0.017 * x[1] + 85.852)
定义约束条件
g1 = -0.8601 + 0.0542 * x[0] + 0.0018 * x[1] - 3
g2 = -1 * (-0.8601 + 0.0542 * x[0] + 0.0018 * x[1])
g3 = -91.4094 + 0.0688 * x[0] + 0.0027 * x[1] + 85
g4 = 91.4094 - 0.0688 * x[0] - 0.0027 * x[1] - 100
g5 = 90.3490 + 0.0841 * x[0] + 0.0030 * x[1] - 100

<code>g6 = -1 * (90.3490 + 0.0841 * x[0] + 0.0030 * x[1])</code>
<code># todo</code>
<code>out["F"] = np.column_stack([f1, f2])</code>
<code>out["G"] = np.column_stack([g1, g2, g3, g4, g5, g6])</code>
<code>problem = MyProblem()</code>
<code># 定义遗传算法</code>
<code>algorithm = NSGA2(pop_size=40, n_offsprings=10, elimdate_duplicates=False)</code>
<code>res = minimize(problem, algorithm, ('n_gen', 300), verbose=False)</code>
<code>plt.scatter(res.F[:, 0], res.F[:, 1], marker="o", s=10)</code>
<code>plt.grid(True)</code>
<code>plt.show()</code>
<code>X = res.X</code>
<code>F = res.F</code>
<code>xl, xu = problem.bounds()</code>
<code>plt.figure(figsize=(7, 5))</code>
<code>plt.scatter(X[:, 0], X[:, 1], s=30, facecolors='none', edgecolors='r')</code>
<code>plt.xlim(xl[0], xu[0])</code>
<code>plt.ylim(xl[1], xu[1])</code>
<code>plt.title("Design Space")</code>
<code>plt.show()</code>
<code>approx_ideal = F.min(axis=0)</code>
<code>approx_nadir = F.max(axis=0)</code>
<code>nF = (F - approx_ideal) / (approx_nadir - approx_ideal)</code>
<code>fl = nF.min(axis=0)</code>
<code>fu = nF.max(axis=0)</code>
<code>weights = np.array([0.5, 0.5])</code>
<code>from pymoo.decomposition.asf import ASF</code>
<code>decomp = ASF()</code>
<code>i = decomp.do(nF, 1/weights).argmin()</code>
<code>print("Best regarding ASF: Point \ni = %s\nF = %s" % (i, F[i]))</code>
<code>print("Best regarding ASF: Point \ni = %s\nX = %s" % (i, X[i]))</code>

plt.figure(figsize=(7, 5))
for i in range(len(F)):
F[i][1] = -1 * F[i][1]
plt.scatter(F[:, 0], F[:, 1], s=30, facecolors='none', edgecolors='blue')
plt.scatter(F[i, 0], F[i, 1], marker="x", color="red", s=200)
plt.title("Objective Space")
plt.xlabel(r"Filtration resistance")
plt.ylabel(r"Filtration efficiency")
plt.show()

6. 最终进行敏感性分析作图代码

import matplotlib.pyplot as plt
import numpy as np
robutness = [0.95, 0.96, 0.97, 0.98, 0.99, 1, 1.01, 1.02, 1.03, 1.04, 1.05]
a_1_1_4 = [2.6394, 2.6612, 2.6829, 2.7045, 2.7262, 2.7479, 2.7696, 2.7913, 2.8129, 2.8346, 2.8563]
a_1_1_2 = [1.9697, 1.9805, 1.9913, 2.0022, 2.0130, 2.0239, 2.0347, 2.0455, 2.0564, 2.0672, 2.0781]
b_1_1_4 = [28.3544, 28.2514, 28.1484, 28.0454, 27.9424, 27.8394, 27.7365, 27.6335, 27.5305, 27.4275, 27.3245]
b_1_1_2 = [33.5860, 33.5101, 33.4343, 33.3584, 33.2826, 33.2067, 33.1308, 33.0550, 32.9791, 32.9033, 32.8274]
h = np.arange(0,11)
fig, axs = plt.subplots(2, 2)
axs[0,0].set_ylim([0,4])
axs[0,0].plot(h, robutness, label = r"\$x_1\$",color='tab:pink',marker='D', markersize=4)
axs[0,0].plot(h, a_1_1_2, label = r"\$y_1\$",color='cornflowerblue',marker='^', markersize=4)
axs[0,0].set_xlabel(r'fluctuate the coefficient of \$x_1\$')
axs[0,0].set_title(r"(a).\$x_1\$ = 20, \$x_2\$ = 1000, \$y_1\$", fontsize=10)
axs[0,1].set_ylim([0,4])
axs[0,1].plot(h, robutness, label = "Fluctuation factor",color='tab:pink',marker='D', markersize=4)
axs[0,1].plot(h, a_1_1_4, label = "Dependent variable",color='cornflowerblue',marker='^', markersize=4)
axs[0,1].set_xlabel(r'fluctuate the coefficient of \$x_1\$')
axs[0,1].set_title(r"(b).\$x_1\$ = 40, \$x_2\$ = 800, \$y_1\$", fontsize=10)
axs[1,0].set_ylim([0,35])
axs[1,0].plot(h, robutness, label = "",color='tab:pink',marker='D', markersize=4)
axs[1,0].plot(h, b_1_1_2, label = "",color='cornflowerblue',marker='^', markersize=4)
axs[1,0].set_xlabel(r'fluctuate the coefficient of \$y_1\$')
axs[1,0].set_title(r"(c).\$x_1\$ = 20, \$x_2\$ = 1000, \$w_1\$", fontsize=10)

axs[1,1].set_ylim([0,35])
axs[1,1].plot(h, robutness, label = "",color='tab:pink',marker='D', markersize=4)
axs[1,1].plot(h, b_1_1_4, label = "",color='cornflowerblue',marker='^', markersize=4)
axs[1,1].set_xlabel(r'fluctuate the coefficient of \$y_1\$')
axs[1,1].set_title(r'(d).\$x_1\$ = 40, \$x_2\$ = 800, \$w_1\$"', fontsize=10)
axs[0,1].legend(loc='upper left', bbox_to_anchor=(1.05,1.0),borderaxespad = 0.) ##设置 ax4 中 legend 的位置， 将其放在图外
fig.tight_layout()
plt.show()