


中国研究生创新实践系列大赛

“华为杯”第十七届中国研究生

数学建模竞赛

学　　校	北京大学
参赛队号	20100010031
队员姓名	1.万芷萱
	2.倪效龙
	3.梁朝茜

中国研究生创新实践系列大赛

“华为杯”第十七届中国研究生

数学建模竞赛

题 目 汽油辛烷值损失预测模型的构建及应用

摘 要:

汽油精制处理过程普遍面临着环保要求和质量要求的两难困境。如果在保证硫含量和烯烃含量达到环保要求的基础上，有效减少汽油辛烷值的损失将产生较大的经济和社会效益。本文按照数据预处理、影响汽油辛烷值损失的操作变量的提取，建立和评价辛烷值损失预测模型，对精制过程中的操作进行优化四个步骤，建立了一个完整的辛烷值损失预测模型和优化方案，同时按照题目要求解决了相关问题。

对于问题一：我们根据操作经验确定的取值范围和 20/80 原则，对非合理数据和数据大量缺失的数据列设为空值，并运用线性插值法和平均值法对空缺数据进行补充，然后我们将处理后的 285 号和 313 号数据加入到包含 325 个样本的数据中。但是包含 325 个样本的数据仍然存在数据缺失、异常等情形，为了提高数据的有效性和统一性，我们对包含 325 个样本的数据重复上述操作。同时根据拉依达准则将误差较大的数据设为空值。最后，我们通过独创性的反距离权重插值法对 305 个空缺数据进行合理补充，获得了较为合理和完整的数据；

对于问题二：由于精制过程的复杂性和操作设备的多样性，操作变量具有高维、非线性相关和高度耦合的特点，我们利用五种线性或非线性的降维方法（算法）对操作变量进行特征提取。本文采用的线性降维方法为逐步回归法、WEIGHT-PCA 法、LarsLasso 算法和 ElasticNet 算法，非线性降维方法为 RandomForest 算法。通过对比分析五种方法（算法）提取出的操作变量集，提取出了六个具有独立性和代表性的最优操作变量；

对于问题三：根据数据和主要变量的特性，本文采用了 SVR 回归、BP 神经网络、KNN 回归、RandomForest 回归和 MLP 神经网络五种非线性回归模型建立辛烷值损失预测模型，同时将 75%的数据设定为训练集、25%为测试集对各个模型进行训练和检验。我们综合对比分析各模型的测试集的均方误差、均方根误差和平均绝对误差以及拟合曲线来判断模型的准确性和拟合效果。最后通过对比和分析五种模型的算法原理、适用条件、拟合效果和综合误差，最终选取了具有高精度和高稳定性的 SVR 模型作为本文的辛烷值损失预测的主要模型；

对于问题四：由于样本中操作变量之间的非线性与强耦合性的关系，非线性模型很难直接通过模型参数获取目标值优化与操作变量调整值之间的联系。因此针对本文采用的 SVR 模型，我们构造了全部操作变量数值组合的方式对预测值进行优化，本文共测试了 444600 个组合的辛烷值损失的变化情况。根据操作变量范围、最大单次调整数和主要操作变量构建变量数值组合，利用 SVR 回归模型对所有可能操作动作中进行分析，最后选择损

失量最小的变量数值组合作为最优动作。根据最优动作优化后，97%的样本的辛烷值损失减少，平均辛烷值损失减少量为 46.49%。

对于问题五：313 号样本基于最优动作的优化后，辛烷值损失值下降 32.80%。我们运用三维图从时间变化和操作变量变动两个维度对 313 号样本的产品硫含量、RON 损失值的变化轨迹进行分析和展示。

关键词：特征提取，辛烷值损失预测，逐步回归法，WEIGHT-PCA 法，LarsLasso 算法，ElasticNet 算法，RandomForest 算法，CVR 模型，神经网络

目录

一：问题重述

1.1 问题背景

汽油燃烧产生的尾气会造成大气污染，主要原因是其中的硫经过燃烧形成有毒有害的大气污染物二氧化硫、三氧化硫等，其中的烯烃属于挥发性有机物即 VOC，经过高温挥发到大气中，加速臭氧的形成，造成大气污染。因此为了降低大气污染，各国都对降低汽油中硫和烯烃的含量提出了更高的要求和标准。下图 1 展示了我国和欧盟的汽油质量的变化，各国都在逐步降低汽油中的硫和烯烃含量，从而达到保护大气环境的目标。

由于我国主要依靠进口获取原油，且大部分进口的原油中以硫为代表的杂质含量较

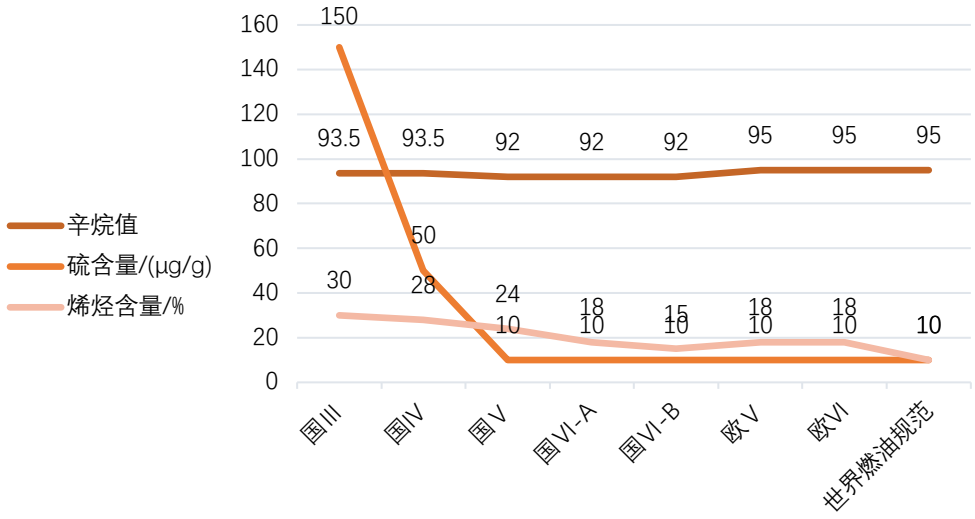


图 1-1:我国和欧盟汽油质量标准

大，为了有效利用和降低石油带来的大气污染，我国主要采用催化裂化技术将原油转换为可利用的汽油和柴油，这一过程有效提高了原油的利用率，但同时也使原油中含有较高的硫和烯烃。所以，我国将在催化裂化技术的基础上对汽油进行进一步的精制处理，以满足汽油质量和环保的要求。现有技术在对催化裂化汽油进行精制（即脱硫和降烯烃）的过程中，普遍降低了汽油辛烷值。而汽油辛烷值是反映汽车燃烧性能最重要的指标，汽油辛烷值的降低将显著降低汽油的燃烧效率。

因此，在汽油精制处理过程中，普遍面临着环保要求和质量要求的两难困境。如果在保证降低硫含量和烯烃含量达到环保要求的基础上，有效减少汽油辛烷值的损失将产生较大的经济和社会效益。首先，减少汽油辛烷值的损失将减少不正常燃烧，震爆、耗油及行驶无力等现象的出现，提高汽油的抗爆性和动力性；其次，减少汽油辛烷值的损失可以提高汽油利用率，进一步促进环境保护。

综上，我们很有必要根据现有汽油精制技术建立辛烷值损失模型并提出先进的、可行的汽油精制技术操作的优化方案，为减少汽油精制技术中的辛烷值损失提供科学的理论依据和优化后的操作指南。

1.2 问题重述

问题一：数据处理

在数据获取的过程中，由于装置设置、采集操作等环境和设备的限制，可能导致原始数据出现数据缺失、异常值等问题。针对部分变量存在少量空缺值和全部空缺值的情况，需要采用不同的方法对数据进行剔除或者补充。针对部分变量出现异常值的情况，需要根据工艺要求、操作经验和数据分布原则确定变量范围，从而对数据进行清洗和筛选。对原始数据进行剔除、补充、清洗等处理后的数据集将作为模型设定和模型验证使用的数据。

问题二：变量选择

在本题所提供的数据中，由于工业过程的复杂性和操作设备的多样性，操作变量数量较多，且操作变量之间具有强耦联的关系，导致无法利用模型进行较好的分析和预测自变量和因变量的关联关系。所以本题需要通过科学和合理的降维方法对变量进行选择，使变量在尽可能的包含所有信息的情况下，减少变量个数，有利于模型的建立、拟合和预测。

问题三：模型建立

利用问题一处理后的数据和问题二筛选的具有代表性和独立性的变量建立辛烷值损失预测模型，该模型将从原料性质、吸附剂性质、产品性质和操作变量等方面阐述精制过程中辛烷值损失的影响因素，然后将利用现有数据对模型预测的精确度和可信度进行分析和评价。

问题四：敏感性分析

根据本文建立的辛烷值损失预测模型，在保持非操作变量不变的情况下，分析辛烷值损失下降对应的主要操作变量的变化程度，从而指导精制过程中减少辛烷值损失的操作方案，为精制过程的优化提供理论估计范围。

问题五：模型展示

由于汽油还需要符合国家的环保要求，在精制过程中，硫含量也是一个非常值得关注的变量，需要清晰和直观的展示在操作变量优化调整过程中辛烷值和硫含量的变化轨迹，使汽油精制过程后的汽油满足质量和环保的双重要求。

二：问题分析

对于问题一：我们首先需要确定需要处理针对的各类数据的各类特征及相应的预处理方法。首先我们需要处理 285 号和 313 号样本数据中的操作变量：（1）由于本文所研究的石化企业的催化裂化汽油精制脱硫装置已运行 4 年，积累了关于精制过程中相关操作变量的取值范围的经验，我们根据操作经验确定的取值范围，对非合理数据设置为空值，同时根据变量范围判断数据中的 0 值是否为合理数据，若当前列范围跨越 0（及 min<0<max），则保留 0 数据，否则设为空值；（2）根据 80%原则（Bijlsma et al.2006），保留具有 80%的数据量的数据列，将具有 20%的数据量缺失的数据列设为空值；（3）由于 285 号和 313 号样本均是由 40 个不同时间测得的数据构成，由于测量时间间隔为 3 分钟或 6 分钟，间隔时间较短，可以认为数据变化符合线性变化，利用线性插值法对空缺数据进行补充；（4）以 285 号和 313 号辛烷值数据测定的时间点为基准时间，取其前 2 个小时的操作变量数据的平均值作为对应辛烷值的操作变量数据。

根据题目要求，我们需要将处理后的 285 号和 313 号数据加入到包含 325 个样本的数据中。然而包含 325 个样本的数据仍然存在数据缺失、异常等情形，为了提高数据的有效性和统一性，需要对包含 325 个样本的数据重复如上（1）（2）的数据处理操作。（3）在通常情况下，大样本数据分布满足正态分布，根据拉依达准则（3σ 准则），误差值超过 3σ 的数据被认为是含有较大误差的数据，我们根据 3σ 原则删除非操作变量中的误差较大的数据；（4）针对数据中仍存在空缺值的情况，本文利用反距离权重法，我们将所有列均为

非空正常值中的 150 个完整样本保留并进行归一化处理，对于具有缺失值的每一个样本，在归一化后，获取当前样本的缺失值变量名，删除 150 个完整样本的对应列，之后求该样本与 150 个完整样本的欧式距离，利用该欧氏距离倒数的平方作为权重，求解 150 个样本在缺失值变量下的算数平均值并赋值给当前样本以缺失数据所在行与完整数据行的欧氏距离作为权重，求解 166 个空缺数据的权重平均值对空缺数据进行补充。

对于问题二：由于精制过程的复杂性和操作设备的多样性，操作变量数量较多，且操作变量之间具有非线性和强耦联的关系，如果将所有操作变量均纳入模型中，将导致模型无法较好进行较好的分析和预测自变量和因变量的关联关系。所以本题需要通过科学和合理的降维方法对变量进行选择，使变量在尽可能的包含所有信息的情况下，减少变量个数，有利于模型的建立、拟合和预测。一般来说，通过特征提取降维是指通过研究特征之间的关系，从原始特征集中提取出能代表原始特征集的子集，从而找出隐藏在高维观测数据中有意义的的低维结构。降维方法分为线性降维和非线性降维，由于本文的操作变量间的关系较为复杂，难以准确的判断变量间的关系，所以本文采用线性和非线性降维中具有代表性的方法，线性降维方法采用逐步回归法、PCA 法、Lars-LarsLasso 法、ElasticNet 算法，非线性降维方法采用 RandomForest 算法。本文将分析和总结不同方法提取出的操作变量集，取五个操作变量集的交集作为本文的主要操作变量。（1）模型一我们选用的是逐步回归法，对于存在多重共线性的数据，逐步回归法是一种有效的变量筛选方法（李松臣等，2008）；（2）模型二我们选用的是 WEIGHT-PCA 法对多维变量系统进行降维处理，找出隐藏在高维观测数据中有意义的低维结构，并根据每个主成分中的原始变量个数进行按权重分配，最后选取出能较完整的包含完整原始变量的主要变量；（3）模型三我们选用的是 LarsLasso 算法，LarsLasso 算法可以将大量的冗余变量去除，只保留与因变量最相关的解释变量，简化模型的同时却保留数据集中最重要的信息；(4)模型四我们选用的是 ElasticNet 算法，ElasticNet 在 LarsLasso 加入正则化的惩罚范数的基础上，在变量选择上具有以部分变量作为一个整体同时被选中或剔除的特性，因此，ElasticNet 的变量筛选和降维的效率更高，适合解决本文对高维变量的降维问题；（5）模型五我们用的是 RandomForest 算法，随机森林模型是在以决策树为基学习器构建的 Bagging 集成的基础上，进一步在决策树的训练过程中引入随机属性选择的模型。本文通过运用随机森林模型对高维的数据进行筛选和分类，有助于了解变量间的关系和变量对对于因变量的影响。通过这五个模型分别筛选出对因变量最具有代表性的主要操作变量，通过比较和分析不同模型的应用范围和结论的误差，选取其中最优的结果作为本文的主要操作变量并纳入辛烷值（RON）的损失的预测模型进行分析，从而得出具有合理性和准确性的辛烷值（RON）损失预测模型。

对于问题三：我们利用了问题一处理后的数据和问题二筛选的具有代表性和独立性的变量建立辛烷值损失预测模型，然后通过现有数据对模型预测的精确度和可信度进行分析和评价。基于数据特征较多且稀疏和特征之间呈现高度非线性和强耦联的样本特点，本文侧重于采用区别与传统回归模型的机器学习非线性回归模型，以提高变量之间关系的解释性、模型的拟合性和预测值的准确性。本文选取出了支持向量回归（SVR）、BP 神经网络、K 近邻（KNN）回归、随机森林回归和多层感知器（MLP 神经网络）五种模型分别建立辛烷值（RON）损失的预测模型：（1）支持向量回归（SVR）模型将线性不可回归的样本点通过升维实现线性化，使得样本在这个特征空间内线性可分，可以很好地拟合非线性趋势；（2）BP 神经网络模型基于误差反向传播的神经网络，能够自组织、自适应和自学习，可以有效的对大规模的数据进行并行处理；（3）KNN 回归模型基于实例进行自学习，能通过建立向量空间模型对于变量间的非线性关系进行拟合；（4）随机森林模型基于决策树的高度灵活的机器学习算法，对于基于特征选取的回归问题具有很好的拟合效果；（5）MLP 网络模型通过前向传播得到误差， 再把误差通过反向传播实现权重值 w 的修正，最终得

到最优结果的模型。我们将 75%的数据作为训练集、25%的数据作为测试集进行模型的训练和检验。在测试集上，本文通过对比和分析五种模型的算法原理、适用条件、拟合结果和误差，最终选取了具有高精度和算法稳定的 SVR 模型作文本文的主要模型。

对于问题四：由于样本中操作变量之间的非线性与强耦合性的关系，采用非线性模型很难直接通过模型参数获取目标值优化与操作变量调整值之间的联系。因此对于优化非线性模型的预测值，我们采用构造全部操作变量数值组合的方式，测试不同组合的目标值变化情况。根据附件四中的操作变量范围与最大单次调整数和第二问中获取的主要操作变量，构建变量数值组合，进而利用第三问的回归模型对所有可能操作动作中最优的操作动作进行分析，预测辛烷值（RON）的损失量，并选择损失量最小的变量数值组合作为最优动作。

对于问题五：由于涉及多个操作变量，我们假设在经过单位时间后，所有非操作变量可以同时改变相应的Δ值。对于非操作变量中的产品硫含量，我们逐步将其从原始数据的 3.2μg/g 提升至 5μg/g。我们利用问题四获得的最优动作，以及附件四中各操作变量的Δ值，计算出完成全部操作变量调整的最长时间。我们从时间变化和操作变量变动两个维度对产品硫含量、RON 损失值的进行分析和展示。

问题一至问题五的具体解答思路如图 2-1:

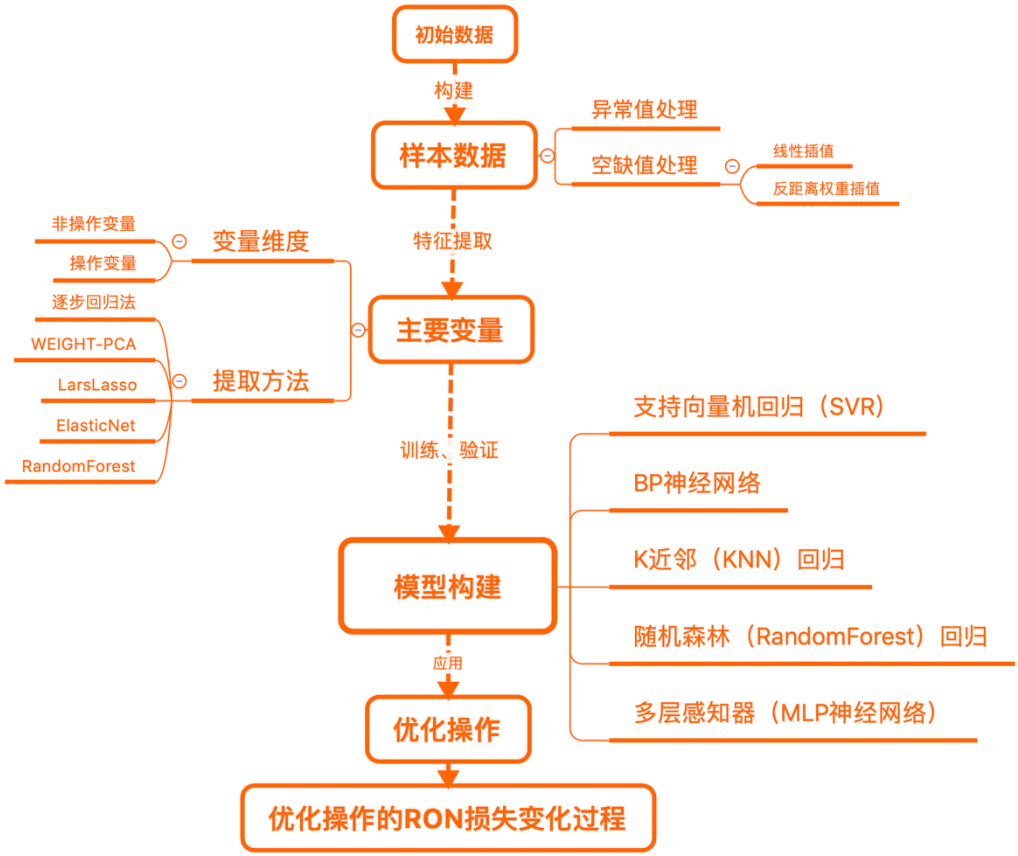


图 2-1 问题一至问题五的解题思路流程图

三：模型假设

（1） 非操作变量数据符合正态分布；
（2） 操作变量数据在测量时间间隔间符合线性变化；
（3） 取辛烷值测定前 2 个小时的操作变量数据的平均值作为对应辛烷值的操作变量数据不影响数据有效性；
（4） 在单位时间内，选取的主要操作变量可以同时变化相应单次最大可改变值

四：符号说明

符号	说明
y_i	因变量，第i个样本的辛烷值（RON）的损失
x_{ij}	第i个样本的第j个操作变量的值
n	样本数
p	操作变量数
Δ_j	第j个操作变量的单次最大可改变值
N_j	第j个操作变量的选择粒度与Δ_j的比值
$timestep$	时间序列坐标
k	邻近值个数
ω_i	权重值
b	偏置项的阈值
ε	隔离带距离
M	隐藏层神经元的个数
n	输入层神经元个数
m	输出层神经元个数
a	区间[0-10]的常数
O	时间复杂度
N	样例个数
Ti	训练集抽取的轮数
D	空间特征
f	激活函数
w	神经网络的权重

表 4-1 符号说明

五：问题一模型的建立与求解

5.1 问题描述与分析

在数据获取的过程中，由于装置设置、采集操作等环境和设备的限制，可能导致原始数据出现数据缺失、异常值等问题。经过统计，285 号和 313 号样本原始数据共有 28346 条数据，325 个样本数据共有 19504 条数据。对于不同类型的原始数据出现的少量空缺、大量空缺、异常、误差较大等数据问题，需要采用不同的方法对数据进行剔除或者补充。问题一的总体思路如下：

图 5-1 285 号和 313 号数据处理流程图

本文根据操作经验、数据分布特点、拉依达准则、权重法等数据处理方法对数据进行清洗和筛选，从而将剔除、补充、清洗等处理后的数据集作为模型设定和模型验证使用的数据。

5.2 285 和 313 号数据的处理

5.2.1 原始数据的统计性分析

问题一要求需要先对 285 号和 313 号数据进行预处理，由于两个样本数据分别是从 2017 年 7 月 17 日 06:03 到 08:00 和 2017 年 5 月 15 日 06:03 到 08:00 以 3 秒或 6 秒为时间间隔测量，数据量较大，本文对数据进行了统计性分析并截取部分进行展示，说明原始数据部分重要特性，见下表 5-1。

数据样本	285 号	313 号
时间范围	2017-7-17 06:03-08:00	2017-05-15 06:03:00-08:00
数据量	14200	14200

平均值（截取部分数据）		
S-ZORB.CAL_H2.PV	0.2734	0.2619
S-ZORB.PDI_2102.PV	24.2082	17.1835
S-ZORB.PT_2801.PV	2.5289	2.4171
S-ZORB.FC_2801.PV	855.8825	850.3009
S-ZORB.TE_2103.PV	421.5093	424.9686
S-ZORB.FT_1503.DACA.PV	2200.7890	1943.6910
ZORB.FT_1504.TOTALIZERA.PV	5984749.0000	2154164.0000
标准差（截取部分数据）		
S-ZORB.CAL_H2.PV	0.0001	0.0036
S-ZORB.PDI_2102.PV	0.0180	0.5939
S-ZORB.PT_2801.PV	0.0002	0.0043
S-ZORB.FC_2801.PV	0.2194	12.6119
S-ZORB.TE_2103.PV	0.0023	2.3226
S-ZORB.FT_1503.DACA.PV	5.5243	736.7886
S-ZORB.FT_1504.DACA.PV	0.3036	33.7559

表 5-1 285 号和 313 号原始数据的统计性分析

如上表所示，285 号和 313 号原始数据的部分操作变量的标准差较大，说明操作变量数据可能存在空缺或异常问题，需要利用科学的数据处理方法对数据进行剔除、清洗或补充。

5.2.2 删除明显不合理的数据

依照附件 4 提供的操作变量的操作范围，对 285 号和 313 号原始数据中的不合理数据进行筛选，如果数据超过操作范围（即大于操作范围上限或小于操作范围下限）则将该数据设置为空值。同时根据变量范围判断数据中的 0 值是否为合理数据，如果当前操作范围跨越 0 值（即操作范围下限小于 0 且操作范围上限大于 0），则保留原始数据中的 0 值，否则也将该数据判定为不合理数据，设置为空值。经过如上删除不合理数据的处理后，285 号和 313 号数据中空值情况如表 5-2 和表 5-3。

操作变量	空值数
S-ZORB.FT_1501.PV	40
S-ZORB.FT_1002.PV	40
S-ZORB.SIS_LT_1001.PV	40
S-ZORB.FC_1202.PV	40
S-ZORB.AI_2903.PV	40
S-ZORB.FT_1501.TOTAL	40
S-ZORB.FT_5102.PV	40
S-ZORB.FT_1204.TOTAL	40
S-ZORB.FT_2901.DACA	40
S-ZORB.FC_1104.DACA	40
S-ZORB.FT_2803.DACA	40
S-ZORB.FT_1502.DACA	40

S-ZORB.TEX_3103A.DACA	40
S-ZORB.FT_5102.DACA.PV	40

表 5-2 285 号原始数据删除不合理数据后的空值数

操作变量	空值数
S-ZORB.AT_5201.PV	39
S-ZORB.FT_1501.PV	40
S-ZORB.PT_9403.PV	3
S-ZORB.FT_9402.PV	1
S-ZORB.PDC_2502.PV	20
S-ZORB.FC_2501.PV	4
S-ZORB.FT_1002.PV	40
S-ZORB.SIS_LT_1001.PV	40
S-ZORB.PC_6001.PV	2
S-ZORB.PT_6002.PV	5
S-ZORB.AI_2903.PV	40
S-ZORB.PDC_2607.PV	2
S-ZORB.FT_1501.TOTAL	40
S-ZORB.FT_1204.PV	2
S-ZORB.FT_1204.TOTAL	40
S-ZORB.PC_3101.DACA	1
S-ZORB.FT_2901.DACA	40
S-ZORB.PT_2501.DACA	8
S-ZORB.PT_2502.DACA	10
S-ZORB.FC_2432.DACA	22
S-ZORB.FT_2431.DACA	8
S-ZORB.FC_1104.DACA	40
S-ZORB.FT_2803.DACA	40
S-ZORB.PDI_2801.DACA	3
S-ZORB.PDI_2301.DACA	3
S-ZORB.FT_1502.DACA	40
S-ZORB.BS_LT_2401.PV	14
S-ZORB.PC_2401.DACA	4
S-ZORB.PC_2401B.DACA	3
S-ZORB.PC_2401B.PIDA.SP	3
S-ZORB.PC_2401B.PIDA.OP	4
S-ZORB.PC_2401.PIDA.OP	7
S-ZORB.PC_2401.PIDA.SP	6
S-ZORB.PDT_2409.DACA	2
S-ZORB.FC_2432.PIDA.SP	16
S-ZORB.TE_1603.DACA	9
S-ZORB.TEX_3103A.DACA	40
S-ZORB.AT-0006.DACA.PV	7

S-ZORB.AT-0012.DACA.PV	3
S-ZORB.FT_1204.DACA.PV	2

表 5-3 313 号原始数据删除不合理数据后的空值数

设置为空值的数据将在下面的数据处理过程中被合理补充，最大化现有数据提供的信息对不良数据进行处理。

5.2.3 删除大量缺失数据列

Bijlsma 等人曾对数据处理中的缺失值处理方法提出 20/80 原则，根据 20/80 原则，如果一列数据有 20%的数据量缺失，为了保证数据的有效性，需要删除该列数据。本文依照 20/80 原则对 285 和 313 号原始数据中缺失率超过 20%的数据列设置为空值列，处理的操作变量如表 5-4。

285 号数据设置为空值列的变量	313 号数据设置为空值列的变量
S-ZORB.FT_1501.PV	S-ZORB.AT_5201.PV
S-ZORB.FT_1002.PV	S-ZORB.FT_1501.PV
S-ZORB.SIS_LT_1001.PV	S-ZORB.PDC_2502.PV
S-ZORB.FC_1202.PV	S-ZORB.FT_1002.PV
S-ZORB.AI_2903.PV	S-ZORB.SIS_LT_1001.PV
S-ZORB.FT_1501.TOTAL	S-ZORB.AI_2903.PV
S-ZORB.FT_5102.PV	S-ZORB.FT_1501.TOTAL
S-ZORB.FT_1204.TOTAL	S-ZORB.FT_1204.TOTAL
S-ZORB.FT_2901.DACA	S-ZORB.FT_2901.DACA
S-ZORB.FC_1104.DACA	S-ZORB.PT_2502.DACA
S-ZORB.FT_2803.DACA	S-ZORB.FC_2432.DACA
S-ZORB.FT_1502.DACA	S-ZORB.FC_1104.DACA
S-ZORB.TEX_3103A.DACA	S-ZORB.FT_2803.DACA
S-ZORB.FT_5102.DACA.PV	S-ZORB.FT_1502.DACA
	S-ZORB.BS_LT_2401.PV
	S-ZORB.FC_2432.PIDA.SP
	S-ZORB.TE_1603.DACA
	S-ZORB.TEX_3103A.DACA

表 5-4 285 和 313 号原始数据设置为空值列的操作变量

5.2.4 补充时间连续的缺失值数据

由于 285 号和 313 号两个样本数据分别是从 2017 年 7 月 17 日 06:03 到 08:00 和 2017 年 5 月 15 日 06:03 到 08:00 以 3 秒或 6 秒为时间间隔测量得到的，在较短的时间间隔，操作变量的数据变化可以被认定为线性变化的，同时，为了验证数据变化的线性特点，分别从 285 号和 313 号样本中任意取四个操作变量的变化折线图如下图 5-2 和 5-3 所示。

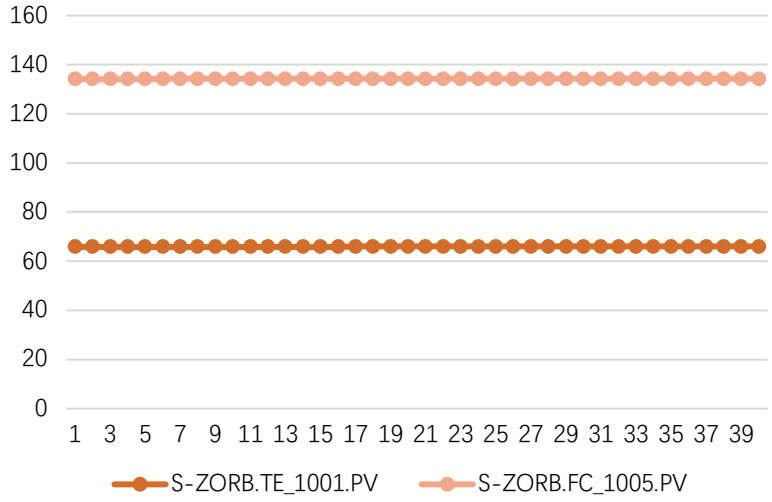


图 5-2:285 号数据任意两个操作变量数据随时间变化的趋势

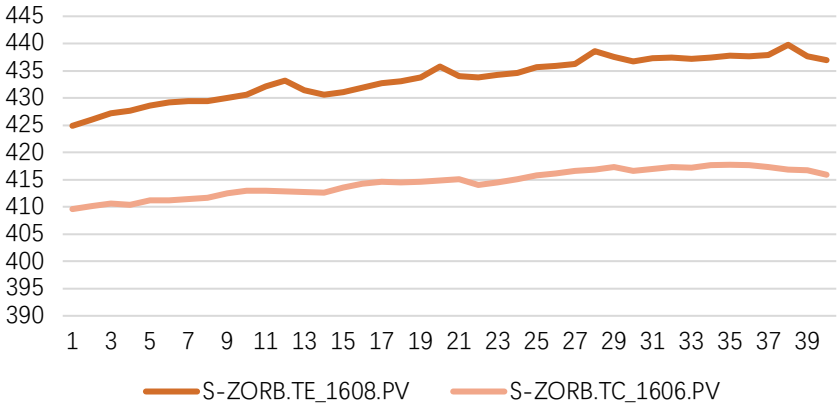


图 5-3 313 号数据任意两个操作变量数据随时间变化的趋势

从如上 285 和 313 号数据任意两个操作变量数据随时间变化的趋势可知，原始数据中操作变量随时间的变化是符合线性的且平滑的，可以对空值进行线性插值处理。

5.3 325 个样本的数据的处理

5.3.1 原始数据的统计性分析

将处理过后的 285 号和 313 号数据添加到 325 个样本的数据中后，该数据的统计性分析部分结果如表 5-5 和 5-6 所示。

名称	非操作变量
时间范围	2020/5/26 8:00:00-2017/4/17 8:00:00
数据量	4875
平均值（截取部分数据）	
硫含量	229.5489
辛烷值 RON	89.7015
饱和烃	52.6762

烯烃	25.3893
芳烃	21.9317
标准差（截取部分数据）	
硫含量	65.7522
辛烷值 RON	0.9501
饱和烃	4.5843
烯烃	4.9549
芳烃	1.8532

表 5-5 325 个样本的数据中非操作性变量的统计性分析

名称	操作变量
时间范围	2020/5/26 8:00:00-2017/4/17 8:00:00
数据量	115050
平均值（截取部分数据）	
S-ZORB.CAL_H2.PV	0.0243
S-ZORB.PDI_2102.PV	3.7657
S-ZORB.PT_2801.PV	0.0449
S-ZORB.FC_2801.PV	76.1802
S-ZORB.TE_2103.PV	2.5406
标准差（截取部分数据）	
S-ZORB.CAL_H2.PV	0.0243
S-ZORB.PDI_2102.PV	3.7657
S-ZORB.PT_2801.PV	0.0449
S-ZORB.FC_2801.PV	76.1802
S-ZORB.TE_2103.PV	2.5406

表 5-6 325 个样本的数据中操作性变量的统计性分析

从统计性分析来看，325 个样本数据均值基本符合操作范围、标准差不存在异常，但是为了样本数据的统计性，本文依旧对 325 个样本的数据进行处理，使每个样本数据的处理相同，且尽可能的利用已知的数据信息对数据进行科学和合理的剔除和补充。

5.3.2 删除明显不合理数据

和 5.2.2 的处理方法相同，依照附件 4 提供的操作变量的操作范围，对 325 个样本的原始数据中的不合理数据进行筛选，如果数据超过操作范围（即大于操作范围上限或小于操作范围下限）则将该数据设置为空值。同时根据变量范围判断数据中的 0 值是否为合理数据，如果当前操作范围跨越 0 值（即操作范围下限小于 0 且操作范围上限大于 0），则保留原始数据中的 0 值，否则也将该数据判定为不合理数据，设置为空值。经过如上删除不合理数据的处理后，285 号和 313 号数据中空值情况如表 5-7.

操作变量	空值数
S-ZORB.FC_2301.PV	145
S-ZORB.AT_5201.PV	147

S-ZORB.FT_9301.PV	4
S-ZORB.FT_1501.PV	288
S-ZORB.FT_5104.PV	126
S-ZORB.FT_9101.PV	134
S-ZORB.FT_9402.PV	1
S-ZORB.FT_1002.PV	137
S-ZORB.FT_1003.PV	4
S-ZORB.FT_1004.PV	19
S-ZORB.SIS_LT_1001.PV	325
S-ZORB.FC_1202.PV	219
S-ZORB.FC_3103.PV	214
S-ZORB.AI_2903.PV	315
S-ZORB.FT_1002.TOTAL	187
S-ZORB.FT_1501.TOTAL	123
S-ZORB.FT_1202.TOTAL	10
S-ZORB.FT_5102.PV	109
S-ZORB.FT_1204.TOTAL	137
S-ZORB.FT_3303.DACA	3
S-ZORB.FT_2901.DACA	137
S-ZORB.PT_2502.DACA	1
S-ZORB.FC_2432.DACA	3
S-ZORB.FT_2303.DACA	10
S-ZORB.FT_2302.DACA	3
S-ZORB.FT_2002.DACA	22
S-ZORB.FC_1104.DACA	307
S-ZORB.FT_2803.DACA	297
S-ZORB.FT_1502.DACA	308
S-ZORB.BS_LT_2401.PV	1
S-ZORB.BS_AT_2402.PV	34
S-ZORB.FT_3702.DACA	54
S-ZORB.TE_2001.DACA	1
S-ZORB.FC_2432.PIDA.SP	3
S-ZORB.TE_1603.DACA	1
S-ZORB.TEX_3103A.DACA	214
S-ZORB.AT-0012.DACA.PV	9
S-ZORB.FT_5102.DACA.PV	134
S-ZORB.CAL.LEVEL.PV	1
S-ZORB.FT_1006.DACA.PV	34
S-ZORB.FT_5204.DACA.PV	84
S-ZORB.FT_1006.TOTALIZERA.PV	14
S-ZORB.FT_1503.TOTALIZERA.PV	2

S-ZORB.FT_1504.TOTALIZERA.PV	2

表 5-7 325 个样本的原始数据删除不合理数据后的空值数

设置为空值的数据将在下面的数据处理过程中被合理补充，最大化现有数据提供的信息对不良数据进行处理。

5.3.3 删除大量缺失列数据

和 5.2.3 的处理方法相同，根据 Bijlsma（2006）等人提出的缺失值处理 20/80 原则，如果一列数据有 20%的数据量缺失，为了保证数据的有效性，需要删除该列数据。本文依照 20/80 原则对 325 个变量的原始数据中缺失率超过 20%的数据列设置为空值列，处理的操作变量如表 5-8。

325 个样本数据设置为空值列的变量
S-ZORBFC_2301PV
S-ZORBAT_5201PV
S-ZORBFT_1501PV
S-ZORBFT_5104PV
S-ZORBFT_9101PV
S-ZORBFT_1002PV
S-ZORBSIS_LT_1001PV
S-ZORBFC_1202PV
S-ZORBFC_3103PV
S-ZORBAI_2903PV
S-ZORBFT_1002TOTAL
S-ZORBFT_1501TOTAL
S-ZORBFT_5102PV
S-ZORBFT_1204TOTAL
S-ZORBFT_2901DACA
S-ZORBFC_1104DACA
S-ZORBFT_2803DACA
S-ZORBFT_1502DACA
S-ZORBTEX_3103ADACA
S-ZORBFT_5102DACAPV
S-ZORBFT_5204DACAPV

表 5-8 325 个样本的原始数据设置为空值列的操作变量

5.3.4 删除误差较大数据

根据拉依达准则（3σ 准则），如果数据的误差值超出 3σ（即剩余误差和平均值的差额大于三倍的标准差）则表明该数据存在较大误差，应该视为异常值予以剔除。3σ 准则假设对被测量变量进行等精度测量，得到 x_1，x_2，……，x_n，算出其算术平均值 x 及剩余误差 $v_i=x_i-x$（i=1,2,...,n），并按贝塞尔公式算出标准误差 σ，若某个测量值 x_b 的剩余误差 v_b（1<=b<=n），满足$|v_b|=|x_b-x|>3\sigma$，则应予剔除。贝塞尔公式如下：

$$\sigma=[\frac{1}{n-1}\sum_{i=1}^{n}v_i^2]^{1/2}=\{[\sum_{i=1}^{n}x_i{}^2-(\sum_{i=1}^{n}x_i)^2/n]/(n-1)\}^{1/2} \quad (1)$$

对于非操作变量，我们根据 3σ准则删除在 3σ范围外的异常数据，删除的非操作变量个数如表 5-9。

非操作变量	删除个数
辛烷值	2
芳烃	1
溴值	4
硫含量	7
辛烷值	2
焦炭	6
焦炭	7
S	5

表 5-9 删除的非操作变量数据个数

5.3.5 补充用反距离权重法计算的缺失值

我们将全部样本中的 150 个所有列均为非空正常值的样本保留，并对每列数据进行如下归一化处理。

$$x=\frac{\mathcal{X}-\mathrm{Min}}{\mathrm{Max}-\mathrm{Min}} \quad (2)$$

对于具有缺失值的每一个样本，在归一化后，获取当前样本的缺失值变量名，删除 150 个完整样本的对应列，之后求该样本与 150 个完整样本的欧式距离。任意两个 n 维向量$a(\mathcal{X}_{11},\mathcal{X}_{12},\dots\dots,\mathcal{X}_{1n})$与$b(\mathcal{X}_{21},\mathcal{X}_{22},\dots\dots,\mathcal{X}_{2n})$间的欧氏距离计算公式为：

$$d_{12}=\sqrt{\sum_{k=1}^{n}(\mathcal{X}_{1k}-\mathcal{X}_{2k})^2} \quad (3)$$

利用该欧氏距离倒数的平方作为权重，求解 150 个样本在缺失值变量下的算数平均值并赋值给当前样本。补充的数据计算如下公式（4），其中X为存在缺失样本，j为缺失值所在列，j_0为需要填补的列，Y_i为完整样本中被删除的列，d_i为X与Y_i的欧式距离。

$$\mathcal{X}=\frac{\sum_{i=0}^{149}\left(\frac{1}{d_i}^2\right)Y_i[j_0]}{\sum_{i=0}^{149}\left(\frac{1}{d_i}^2\right)} \quad (4)$$

根据反距离权重法对缺失值进行科学合理的补充，原始数据经过一系列处理后成为了不具有异常值、误差较大值和无缺失值的数据，有利于后续模型的建立和预测。

六：问题二的模型建立与求解

6.1 问题描述与分析

由于精制过程的复杂性和操作设备的多样性，操作变量数量较多，且操作变量之间具有非线性和强耦联的关系，如果将所有操作变量均纳入模型中，将导致模型无法较好进行较好的分析和预测自变量和因变量的关联关系。所以本题需要通过科学和合理的降维方法对变量进行选择，使变量在尽可能的包含所有信息的情况下，减少变量个数，有利于模型的建立、拟合和预测。一般来说，通过特征提取降维是指通过研究特征之间的关系，从原始特征集中提取出能代表原始特征集的子集，从而找出隐藏在高维观测数据中有意义的的低维结构。目前传统的降维方法分为线性降维和非线性降维，由于本文的操作变量间的关系较为复杂，难以准确的判断变量间的关系，所以本文采用线性和非线性降维中具有代表性的方法，线性降维方法采用逐步回归法、WEIGHT-PCA 法、LarsLasso 算法和 ElasticNet 算法，非线性降维方法采用 RandomForest 算法。本文将分析和总结不同方法提取出的操作变量集，取五个操作变量集的交集作为本文的主要操作变量。具体的问题分析思路如下：

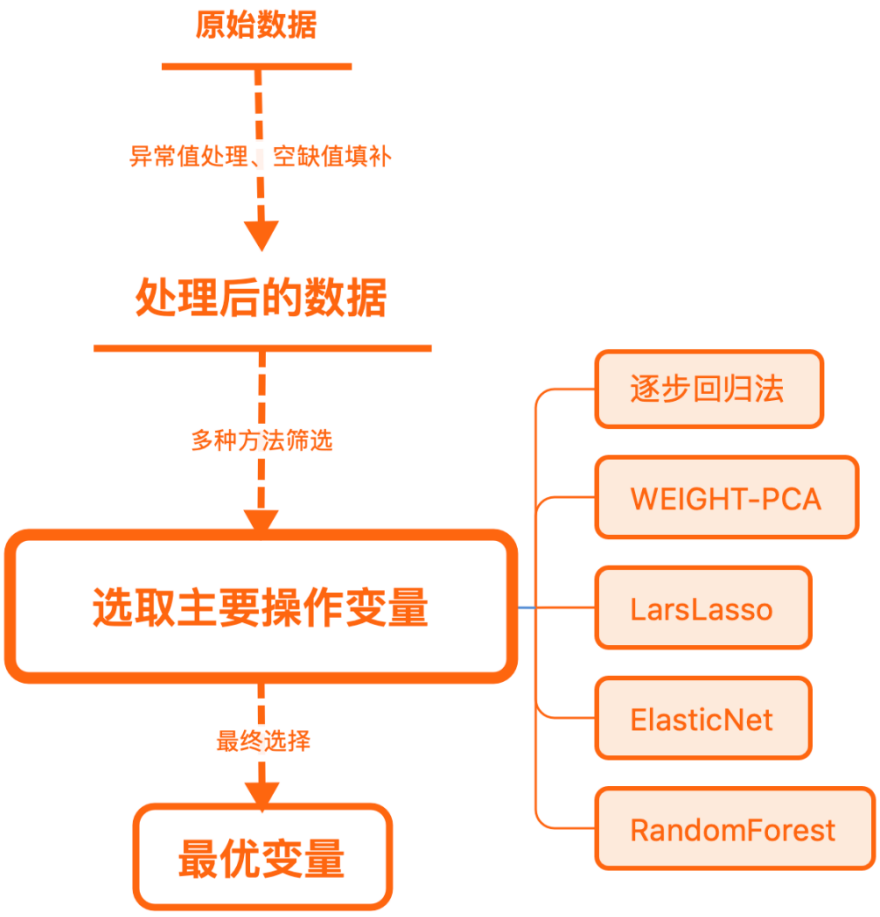


图 6-1：问题二分析流程

数据预处理在问题一已经得到解决。从流程图中可以看出，问题二需要我们对操作变量进行提取，从而达到降维的目的。由于操作变量个数较多，关系较复杂，我们采用多种

模型和方法对变量进行筛选和提取，从中选取最优的操作变量组合，作为后续建模的主要变量。

模型一我们选用的是逐步回归法，通过分析操作变量间的关系，操作变量可能存在线性关系，对于存在多重共线性的数据，逐步回归法是一种有效的变量筛选方法（李松臣等，2008），逐步回归又包括逐步加入和逐步删除两种方式，本文利用逐步加入方式运用逐步回归模型对具有贡献的变量进行筛选；

模型二我们选用的是 WEIGHT-PCA 法，主成分分析法主要通过正交变换，将其分量相关的原随机向量转化成分量不相关的新随机向量，然后对多维变量系统进行降维处理，通过线性映射投影到一个低维特征空间，从而找出隐藏在高维观测数据中有意义的低维结构。本文根据每个主成分的方差贡献，对每个主成分中的原始变量个数进行按权重分配，最后选取出能较完整的包含完整原始变量的主要变量；

模型三我们选用的是 LarsLasso 算法法，在稀疏模型中，LarsLasso 模型得到了普遍认可和应用。LarsLasso 模型可以将大量的冗余变量去除，只保留与因变量最相关的解释变量，简化模型的同时却保留数据集中最重要的信息。但是在 Tibshirani（1997）提出 LarsLasso 模型后，由于缺乏高效的求解算法因而没有引起足够的重视，直到 LARS 算法被提出后，LarsLasso 才逐渐被使用。本文将利用 LarsLasso 模型的变量选择和变量空间降维功能，对主要的操作变量进行筛选；

模型四我们选用的是 ElasticNet 算法，ElasticNet 在 LarsLasso 加入正则化的惩罚范数的基础上，在变量选择上具有以部分变量作为一个整体同时被选中或剔除的特性，ElasticNet 的组效应倾向于将全部高度相关变量作为一个组同时被选中或同时被剔除，如果被选中，则组内变量的回归系数的绝对值（几乎）相等，因此，ElasticNet 的变量筛选和降维的效率更高，适合解决本文对高维变量的降维问题；

模型五我们用的是 RandomForest 算法，随机森林模型是在以决策树为基学习器构建的 Bagging 集成的基础上，进一步在决策树的训练过程中引入随机属性选择的模型。随机森林模型被普遍用于高维数据的筛选和分类。其原理是评估每个变量在随机森林中的每一个任意不相关的决策树上的贡献率，然后取平均值，最后比较不同特征之间的贡献率。本文通过运用随机森林模型对高维的数据进行筛选和分类，有助于了解变量间的关系和变量对对于因变量的影响。

通过这五个模型分别筛选出对因变量最具有代表性的主要操作变量，通过比较和分析不同模型的应用范围和结论的误差，选取其中最优的结果作为本文的主要操作变量并纳入辛烷值（RON）的损失的预测模型进行分析，从而得出具有合理性和准确性的辛烷值（RON）损失预测模型。

6.2 方法原理

6.2.1 逐步回归法确定主要变量

由于本文的操作变量个数较多，各个操作变量之间可能存在线性或非线性的相关关系，本文将首先对操作变量的相关关系进行分析，Pearson 相关系数通常被认为能较好的反映变量间的相关关系，同时本文还将计算操作变量间的 Spearman 相关系数值作为参考和检验。Pearson 相关系数的计算公式如下：

$$\rho_{x_1,x_2}=\frac{n\sum x_1x_2-\sum x_1\sum x_2}{\sqrt{n\sum {x_1}^2-(\sum x_1)^2}\sqrt{n\sum {x_2}^2-(\sum x_2)^2}} \tag{5}$$

Spearman 相关系数是在 Pearson 相关系数的基础上利用两个集合中元素在各自集合中的等级（排名）来计算他们之间的相关性，可以从排序角度来分析两个变量的相关关系。假设两个长度为 n 的向量x_1和x_2，计算x_1和x_2的相关性，需要进行以下步骤：

将两个向对应的元素x_1和x_2转换为在各自列向量中的排序，记为$R(x_{1n})$和$R(x_{2n})$根据公式（6）计算两个列向量x_1和x_2中对应元素$R(x_{1n})$和$R(x_{1n})$之间的差异 d 并相加。

$$d_{x_1,x_2}=\sum_{i=1}^{n}|\mathrm{R}(x_{1n})-\mathrm{R}(x_{2n})|^2 \tag{6}$$

然后根据公式（7）计算出两个列向量之间的相关性。

$$R_{x_1,x_2}=1-\frac{6\times d_{x_1,x_2}}{n\times(n^2-1)} \tag{7}$$

本文的所以操作变量为$\mathcal{X}_1,\mathcal{X}_2,\ldots,\mathcal{X}_n$，用每一个解释变量分别对被解释变量 Y 建立回归模型，得到 n 个回归模型：

$$\begin{cases}y=\partial_{01}+\partial_{11}\mathcal{X}_1+\varepsilon_1\\ y=\partial_{02}+\partial_{12}\mathcal{X}_2+\varepsilon_2\\ \cdots\\ y=\partial_{0n}+\partial_{1n}\mathcal{X}_n+\varepsilon_n\end{cases} \tag{8}$$

对方程（8）进行参数估计，并进行检验，选择通过检验的模型中拟合优度最大的回归模型作为首选模型，假设$\mathcal{X}_1$为对应的首选变量，则首选模型为：

$$y=\partial_{01}+\partial_{11}\mathcal{X}_1+\varepsilon_1 \tag{9}$$

在首选模型中逐个增加其他解释变量，重新进行线性回归。若新增加的解释变量提高了回归的拟合优度，且回归方程中其他参数统计值仍然显著，就在模型中保留该解释变量；若新增加的解释变量没有提高回归方程的拟和优度，就不在模型中保留该解释变量；若新增加的解释变量提高了回归方程的拟和优度，但回归方程中某些参数的数值或符号等受到显著影响， 说明模型中存在多重共线性，将该解释变量同与之相关的其他解释变量进行比较，在模型中保留对被解释变量影响较大的变量，略去影响较小的变量。

6.2.2 WEIGHT-PCA 法确定主要变量

使用主成分分析的前提条件是原始数据各个变量之间有较强的线性相关关系,在 5.2.1 中已对变量间的相关性系数进行计算，本节再次用巴特莱特球性检验(Bartlett test of sphercity)和 KMO(Kaiser-Meyer-Olkin-Measure of Sampling Adequacy)检验方法对操作变量数据是否适用主成分分析进行检验，当巴莱特球性检验差异检验值显著时认为数据适合进行主成分分析;KMO 检验值接近 1,则变量适合进行主成分分析。PCA 法降维首先需要将原始数据标准化，假设原始样本矩阵为：

$$X=(\mathcal{X}_{ij})_{n\times p} \tag{10}$$

数据按照公式（11）进行标准化处理，获得均值为 0，标准差为 1 的标准化数据，

$$Z_{ij}=\frac{\mathcal{X}_{ij}-\overline{\mathcal{X}_J}}{S_j^2} \tag{11}$$

该式中，$\overline{\mathcal{X}_J}$表示第j个指标的平均值，S_j^2表示第j个指标的方差。

原始数据经过处理后得到标准化数据矩阵，计算其对应的相关系数矩阵 R，并计算 R 的特征值和特征向量：

$$R = (r_{ij})_{p\times p} \tag{12}$$

R的特征值将其按大小排列，即为主成分的方差，其大小也描述了对应主成分对原始样本的权重。本文根据主成分方差累计贡献率 60%为临界值确定主成分个数。由于本文将对操作变量进行筛选，本文将每一个主成分方差贡献率作为权重，从而计算每一个主成分中筛选的操作变量个数：

$$\eta = 18 \times \frac{F_i}{\sum_i^q F_i} \tag{13}$$

该式中F_i为第i个成分的方差贡献率，q为将主成分方差累计贡献率的临界值设为 60%确定的主成分数。然后对每个成分中的每个操作变量的系数取绝对值后排序，对应每个成分筛选的操作变量数η，取系数的绝对值较大的前η个操作变量作为主要操作变量。

6.2.3 LarsLasso 算法确定主要变量

已知线性回归模型用向量表示为

$$y = X\beta + \varepsilon \tag{14}$$

$y \in R^n$为响应向量，$\mathcal{X} \in R^{n\times p}$为设计矩阵，$\beta \in R^p$为回归系数向量，$\varepsilon \in N(0,\sigma^2)$为误差向量且全部误差变量独立同分布。LarsLasso 模型是在收到子集选择模型、岭回归估计模型、非负绞刑估计模型和桥回归模型等模型的基础和启发之上被提出的，其通过对回归系数的绝对值之和（即回归系数向量的L_1范数）进行惩罚来压缩回归系数的大小，使绝对值较小的回归系数自动被压缩为 0，从而产生稀疏解和实现变量选择，Tibshirani（1998）提出的基于方程（14）中线性回归模型的 LarsLasso 为：

$$\hat{\beta} = arg\min_{\beta\in R^p} \frac{1}{2}\left|\left[y - X\beta\right]_2^2 + \vartheta \diamond |\beta|\right|_1 \tag{15}$$

其中$\vartheta \geq 0$为可调参数（turning parameter），$\left|\left[y - X\beta\right]_2^2 + \vartheta \diamond |\beta|\right|$ 表示L_2范数。与最小二乘法和岭回归不同，LarsLasso 不具有显性解，其解可通过 Lars 算法进行不断迭代求得，Efron 等提出的 Lars（Least Angle Regression）算法能够高效的求解 LarsLasso 解。Lars 的标准就是要保证当前残差和已入选变量之间的相关系数相等，也就是当前残差在已入选变量的构成空间中的投影，是那些变量的角平分线。当选择了一个变量之后不断吸收新的变量进入，然后调整模型不断吸收新的变量直到当前模型的残差和已入选变量之间的相关系数不相等，则停止迭代。具体算法为：

$$\omega_i = (1_i'(X_i'X_i)^{-1}1_i)^{-\frac{1}{2}}(X_i'X_i)^{-1}1_i \tag{16}$$

$X_i'X_i$就是 Lars 算法在当前回归变量集下的变量选择，Efron 定义了一个向量$\hat{d}$，这个向量的元素是$s_i'w_i$，其中s_i'是入选变量与当前残差的相关系数的符号，也是$\hat{\beta}$的符号，对于没有入选的变量，他们对应在$\hat{d}$中的元素为 0.数学表示为：

$$\beta_j(r) = \widehat{\beta_j} + r\widehat{d_j} \tag{17}$$

直到迭代结束后，得到所有变量的参数估计值，变量参数不为零的参数即被视为经过 Lars-LarsLasso 模型筛选出来的具有贡献的主要参数。

6.2.4 ElasticNet 算法确定主要变量

ElasticNet 是在 LarsLasso 加入正则化的惩罚范数的基础上，在变量选择上具有以部分变量作为一个整体同时被选中或剔除的特性的自动阻效应稀疏模型。组效应是指某些变量作为一个整体被同时选中进而参与模型的构造，即具有变量组选择的效果。组效应倾向于将全部高度相关变量作为一个组同时被选中或同时被剔除，如果被选中，则组内变量的回归系数的绝对值（几乎）相等。ElasticNet 模型通过岭罚实现自动组效应，其数学表示为：

$$\hat{\beta} = arg\min_{\beta \in R^p} \frac{1}{2}\|y - X\beta\|_2^2 + \lambda_1\|\beta\|_1 + \lambda_2\|\beta\|_2^2 \qquad (18)$$

其罚函数由 LarsLasso 模型中的L_1范数和岭罚共同组成，被称作弹性网罚函数。L_1范数使得弹性网具有稀疏性，岭罚使得弹性网具有自动组效应的特性。之后的计算步骤和 LarsLasso 相同。ElasticNet 模型能高效的对变量进行筛选，适合本文中的高维变量的选择问题。

6.2.5 RandomForest 算法确定主要变量

RandomForest 模型是 2001 年由 Leo Breiman 和 Culter Adele 开发的一种基于决策树理论的分类模型数据，该算法是一种有监督的机器学习算法，被广泛应用于分类与回归问题中。随机森林算法的基本分类单元是决策树，该算法实质是一个包含多个决策树的分类器，并且其输出类别由决策树输出类别的众数而定。 该算法运算效率高且能够处理高维（即特征变量多）数据，训练速度快而不会出现过拟合现象。随机森林算法对特征选取具有较好的鲁棒性，无需特征筛选也能得到较高的准确率，因此适用于超高维特征向量空间，同时随机森林算法对异常值和噪声具有较高的容忍度且具有较好的数据推广和范化。随机森林基于 Bootstrap 方法重采样产生多个训练集，采样示意图如下：

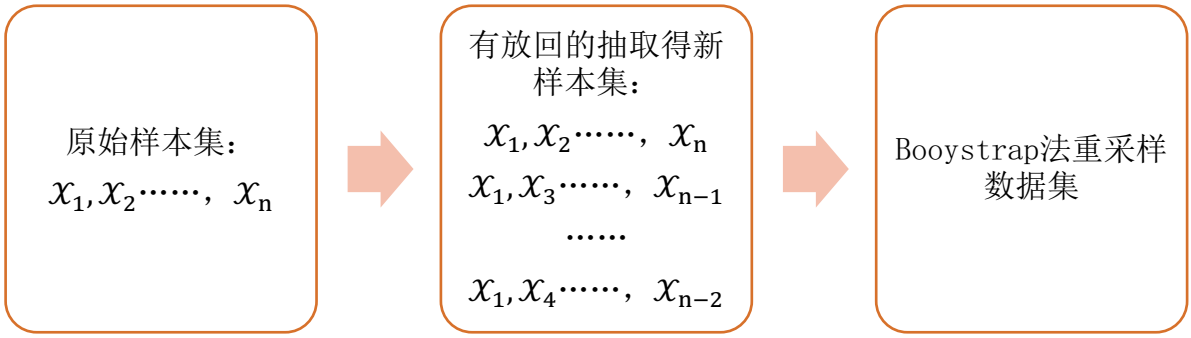


图 6-2：Bootstrap 法重采样示意图

随机森林是以 K 个决策树为基本分类单位，进行集成学习后得到的一个组合分类器。随机森林通过自助聚集法随机选择样本生成决策树，每一棵决策树之间是没有关联的，而且每棵树都会完整生长而不会进行剪枝。在生成树的时候，每个节点的属性变量值都仅 仅在随机选出的少数几个属性子集中产生。通过这两种在数据和属性变量值中的随机性可生

成大量的树，我们称之为随机森林。在得到森林之后，当有一个新的测试样本进入随机森林时，其实就是让每一棵决策树分别进行投票抉择，最终取所有决策树中输出类别最多的那类为分类结果。 最终分类决策可用如下公式表示；

$$H(x) = \arg\max_{y} \sum_{i}^{k} I(h_i(x) = y) \quad (19)$$

在方程（19）中：$H(x)$表示分类组合模型，h_i是单个决策树分类模型，I（·）为示性函数，示性函数是指一个函数使得当集合内有此数时值为 1，当集合内无此数时值为 0。

6.3 模型的求解及分析

6.3.1 逐步回归法的求解结果

(1)相关性分析

由于本文的操作变量个数较多，各个操作变量之间可能存在线性或非线性的相关关系，本文将首先对操作变量的相关关系进行分析，Pearson 相关系数通常被认为能较好的反映变量间的线性相关关系，同时本文还将计算操作变量间的 Spearman 相关系数值作为参考和检验。由于本文的操作变量较多，难以完整和直观的展示每个操作变量的相关性，本文随机选取 5×4 个操作变量的相关性，Pearson 相关系数值及显著性水平见表 5-1， Spearman 相关系数值及显著性水平见表 5-2。

操作变量	S-ZORB.PT_2801.PV	S-ZORB.FC_2801.PV	S-ZORB.TE_2103.PV	S-ZORB.TE_2005.PV
S-ZORB.FT_9201.PV	-0.051	.189**	-.191**	-.128*
S-ZORB.FT_9202.PV	0.004	.380**	-.174**	-0.024
S-ZORB.FT_9302.PV	-0.094	-.651**	-0.107	-.406**
S-ZORB.FT_3301.PV	-.122*	.497**	.281**	.418**
S-ZORB.FT_9402.PV	-.110*	.302**	.192**	.276**

表 5-1：操作变量的皮尔森相关系数

操作变量	S-ZORB.PT_2801.PV	S-ZORB.FC_2801.PV	S-ZORB.TE_2103.PV	S-ZORB.TE_2005.PV
S-ZORB.FT_9201.PV	-0.027**	.209*	-.762**	-.298*
S-ZORB.FT_9202.PV	0.021	.680**	-.181***	-0.004
S-ZORB.FT_9302.PV	-0.001	-.211***	-0.721	-.461***

S-ZORB.FT_3301.PV	-.722***	.247**	.521***	.294**
S-ZORB.FT_9402.PV	-.305*	.102**	.021**	.626**

表 6-2：操作变量的斯皮尔曼相关系数

根据上述两个相关系数表对比，可以发现两种方法计算的相关系数值接近，说明相关系数值是稳健的。通过相关系数值和显著性水平可以发现，部分变量确实存在高度线性相关$R > 0.6$,可以用逐步回归法进行变量筛选。

（2）逐步回归法的变量选择

逐步回归分析湿多元回归分析中的一种方法，但同时，逐步回归分析也可以用于研究多个变量之间相互依赖的关系，从而建立最优或合适的回归模型。逐步回归法中变量的选择主要有逐步加入和逐步删除两种，通过文献的实例可以发现，逐步加入方法的逐步回归法最后选择的变量数较少，逐步删除方法的逐步回归法最后选择的变量较多，可能仍然存在冗余的变量，同时变量间还会仍然存在多重共线性问题。下表是用逐步加入方法的逐步回归模型的结果。

模型	调整后 R 方	标准估算的误差	R 方变化量	F 变化量	自由度 1	自由度 2	显著性 F 变化量	德宾-沃森
1	0.1282	0.2109	0.1309	48.6630	1.0000	323.0000	0.0000	
2	0.1578	0.2073	0.0321	12.3414	1.0000	322.0000	0.0005	
3	0.1779	0.2048	0.0225	8.8837	1.0000	321.0000	0.0031	
4	0.2100	0.2008	0.0342	14.0451	1.0000	320.0000	0.0002	
5	0.2225	0.1992	0.0147	6.1205	1.0000	319.0000	0.0139	
6	0.2329	0.1978	0.0126	5.3128	1.0000	318.0000	0.0218	
7	0.2450	0.1963	0.0142	6.1023	1.0000	317.0000	0.0140	
8	0.2551	0.1950	0.0122	5.3135	1.0000	316.0000	0.0218	
9	0.2698	0.1930	0.0166	7.3566	1.0000	315.0000	0.0070	
10	0.2772	0.1921	0.0094	4.2077	1.0000	314.0000	0.0411	
11	0.2845	0.1911	0.0093	4.2183	1.0000	313.0000	0.0408	2.0644

a. 预测变量：(常量), S-ZORB.FT_9302.PV

b. 预测变量：(常量), S-ZORB.FT_9302.PV, S-ZORB.SIS_TE_2802

c. 预测变量：(常量), S-ZORB.FT_9302.PV, S-ZORB.SIS_TE_2802, S-ZORB.TE_5102.PV

d. 预测变量：(常量), S-ZORB.FT_9302.PV, S-ZORB.SIS_TE_2802, S-ZORB.TE_5102.PV, S-ZORB.TE_1102.DACA.PV

e. 预测变量：(常量), S-ZORB.FT_9302.PV, S-ZORB.SIS_TE_2802, S-ZORB.TE_5102.PV, S-ZORB.TE_1102.DACA.PV, S-ZORB.TE_6001.DACA

f. 预测变量：(常量), S-ZORB.FT_9302.PV, S-ZORB.SIS_TE_2802, S-ZORB.TE_5102.PV, S-ZORB.TE_1102.DACA.PV, S-ZORB.TE_6001.DACA, S-ZORB.TE_5007.DACA

g. 预测变量：(常量), S-ZORB.FT_9302.PV, S-ZORB.SIS_TE_2802, S-ZORB.TE_5102.PV, S-ZORB.TE_1102.DACA.PV, S-ZORB.TE_6001.DACA, S-ZORB.TE_5007.DACA, S-ZORB.PT_1604.DACA

h. 预测变量：(常量), S-ZORB.FT_9302.PV, S-ZORB.SIS_TE_2802, S-ZORB.TE_5102.PV, S-ZORB.TE_1102.DACA.PV, S-ZORB.TE_6001.DACA, S-ZORB.TE_5007.DACA, S-ZORB.PT_1604.DACA, S-ZORB.AT-0002.DACA.PV
i. 预测变量：(常量), S-ZORB.FT_9302.PV, S-ZORB.SIS_TE_2802, S-ZORB.TE_5102.PV, S-ZORB.TE_1102.DACA.PV, S-ZORB.TE_6001.DACA, S-ZORB.TE_5007.DACA, S-ZORB.PT_1604.DACA, S-ZORB.AT-0002.DACA.PV, S-ZORB.TE_2608.DACA
j. 预测变量：(常量), S-ZORB.FT_9302.PV, S-ZORB.SIS_TE_2802, S-ZORB.TE_5102.PV, S-ZORB.TE_1102.DACA.PV, S-ZORB.TE_6001.DACA, S-ZORB.TE_5007.DACA, S-ZORB.PT_1604.DACA, S-ZORB.AT-0002.DACA.PV, S-ZORB.TE_2608.DACA, S-ZORB.FT_9401.PV
k. 预测变量：(常量), S-ZORB.FT_9302.PV, S-ZORB.SIS_TE_2802, S-ZORB.TE_5102.PV, S-ZORB.TE_1102.DACA.PV, S-ZORB.TE_6001.DACA, S-ZORB.TE_5007.DACA, S-ZORB.PT_1604.DACA, S-ZORB.AT-0002.DACA.PV, S-ZORB.TE_2608.DACA, S-ZORB.FT_9401.PV, S-ZORB.LT_3801.DACA
l. 因变量：RON Loss

表 6-3：逐步加入方法的逐步回归模型结果

以偏回归平方和来考虑是否引入新变量，若在 99%的显著性水平下偏回归平方和较大（即变量系数通过显著性检验），则表明可以引入，此时新变量与已选变量不大可能存在相关性。通过这一选择标准，最后经过逐步回归法筛选的操作变量为：S-ZORB.FT_9302.PV, S-ZORB.SIS_TE_2802, S-ZORB.TE_5102.PV, S-ZORB.TE_1102.DACA.PV, S-ZORB.TE_6001.DACA, S-ZORB.TE_5007.DACA, S-ZORB.PT_1604.DACA, S-ZORB.AT-0002.DACA.PV, S-ZORB.TE_2608.DACA, S-ZORB.FT_9401.PV, S-ZORB.LT_3801.DACA（以对 RON Loss 的贡献度降序排列）。

6.3.2 WEIGHT-PCA 法的求解结果

（1）数据适用于 PCA 法的检验

在进行主成分分析之前，我们首先对特征参数数据进行了标准化处理。在此基础上，对特征参数标准化值先进行 KMO 检验及 Barlett 球形检验。结果显示，变量相关矩阵为非正定矩阵，无法进行 KMO 检验及 Barlett 球形检验，这表明我们变量之间相关性过高（该结论也在 5.3.1 中被印证）。但由于在 PCA 选择变量后，我们并不会进行简单的线性回归，综合考虑后我们对这些变量进行保留，直接进行之后的主成分分析操作。

（2）PCA 法的降维效果

PCA 法主要是通过正交变换，将其分量相关的原随机向量转化成分量不相关的新随机向量，然后对多维变量系统进行降维处理，通过线性映射投影到一个低维特征空间，从而找出隐藏在高维观测数据中有意义的低维结构。将 PCA 法处理后的的主成分分别映射到二维和三维中来检验 PCA 法的降维效果，见图 6-3 和图 6-4。

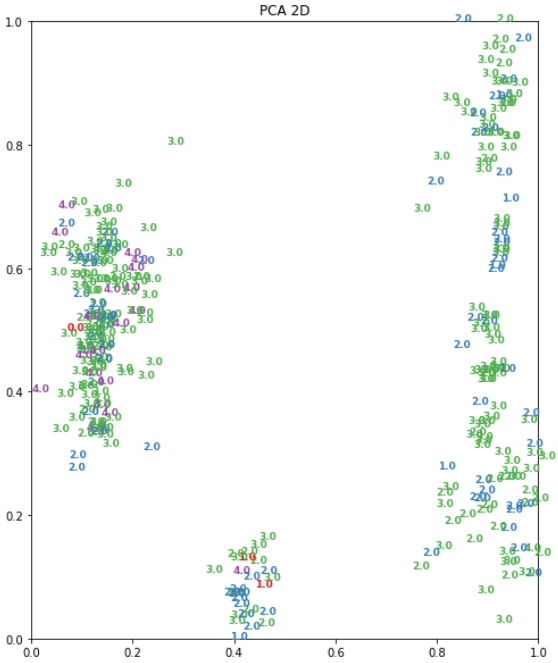


图 6-3 PCA 后的主成分的二维映射

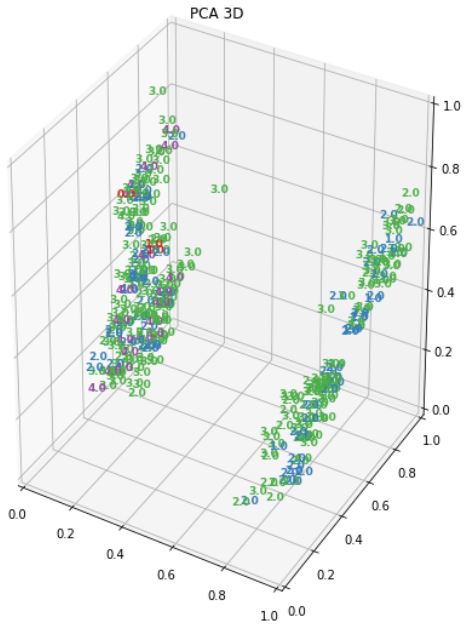


图 6-4 PCA 后的主成分的三维映射

通过上图可以发现，尽管数据自身存在非正定矩阵、相关性较高等问题，但是经过主成分分析后的各个操作变量可以被较为科学和有效的降低维度，说明 PCA 法的降维效果较好。

（3）PCA 中主成分构成结果

根据累计方差贡献率>60%的原则确定 PCA 主成分的个数，PCA 方法确定的各个主成分的方差贡献和累计方差贡献率见表 6-4。

成分	初始特征值		旋转载荷平方和		
	总计	方差百分比	总计	方差百分比	累积 %
1	109.585	32.908	78.634	23.614	23.614
2	38.418	11.537	54.178	16.270	39.884
3	22.813	6.851	33.143	9.953	49.837
4	18.546	5.569	20.767	6.236	56.073
5	13.616	4.089	16.255	4.881	60.954

表 6-4：PCA 的总方差解释结果

同时各个主成分的特征值变化见图 6-5。

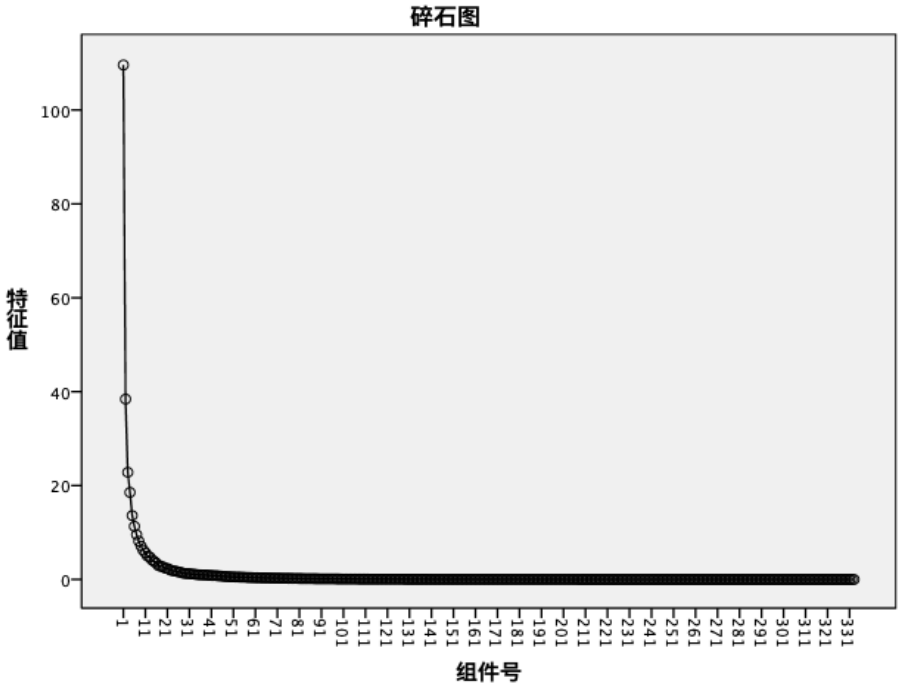


图 6-5：PCA 特征值变化的碎石图

根据上列图标可以发现，前五个成分的初始特征值均大于 1，前五个成分的累计方差贡献率满足>60%的要求，说明前五个成分都能对完整数据的信息进行较好的概括。同时，每个成分的方差百分比 / 累计方差百分比将作为每个主成分的筛选变量数的权重。

（4）WEIGHT-PCA 法的变量选择

由于对非操作性变量（共计 12 个）进行完整保留，由于题目要求模型中的变量不大于 30 个，所以筛选出的操作性变量个数最大为 18 个。本题通过 PCA 进行参数选择的的解决思路是将每个成分的方差百分比 / 累计方差百分比作为每个主成分的筛选变量数的权重，对每个主成分中需要筛选的操作变量的个数进行分配和确定。第一、二、三、四和五主成分的中筛选的操作变量的个数分别为 7、5、3、2 和 1 个。

本文将各个主成分中的操作变量的系数的绝对值进行排序，分别选取前 7、5、3、2 和 1 个的操作变量进行提取，结果如下表。

	第一成分	第二成分	第三成分	第四成分	第五成分
S-ZORB.FC_1101.TOTAL	0.9712				
S-ZORB.FT_9001.TOTAL	0.9706				
S-ZORB.FT_9202.TOTAL	0.9735				
S-ZORB.FT_5201.TOTAL	0.9697				
S-ZORB.FT_1001.TOTAL	0.9702				
S-ZORB.FT_1004.TOTAL	0.9702				
S-ZORB.FT_9302.TOTAL	0.9885				
S-ZORB.FT _1503.TOTALIZERA.PV		-0.9005			
S-ZORB.FT		-0.8875			

_1504.TOTALIZERA.PV				
S-ZORB.FT_3302.DACA	0.8771			
S-ZORB.TE_3111.DACA	0.8779			
S-ZORB.PT_7503.DACA	0.8784			
S-ZORB.TE_3101.DACA		0.9291		
S-ZORB.TE_7506B.DACA		0.9362		
S-ZORB.PT_5201.DACA		-0.9396		
S-ZORB.PT_1102.DACA			-0.8293	
S-ZORB.PT_2801.PV			-0.8068	
S-ZORB.TE_2401.DACA				-0.6560

表 6-5：WEIGHT-PCA 参数选择的结果

根据 WEIGHT-PCA 法参数选择的操作变量共有 18 个，分别是 S-ZORB.FC_1101.TOTAL 、 S-ZORB.FT_9001.TOTAL 、 S-ZORB.FT_9202.TOTAL 、 S-ZORB.FT_5201.TOTAL 、 S-ZORB.FT_1001.TOTAL 、 S-ZORB.FT_1004.TOTAL 、 S-ZORB.FT_9302.TOTAL 、 S-ZORB.FT_1503.TOTALIZERA.PV 、 S-ZORB.FT_1504.TOTALIZERA.PV、S-ZORB.FT_3302.DACA、S-ZORB.TE_3111.DACA、S-ZORB.PT_7503.DACA 、 S-ZORB.TE_3101.DACA 、 S-ZORB.TE_7506B.DACA 、 S-ZORB.PT_5201.DACA 、 S-ZORB.PT_1102.DACA 、 S-ZORB.PT_2801.PV 和 S-ZORB.TE_2401.DACA。

6.3.3 LarsLasso 算法的求解结果

（1）LarsLasso 算法的参数设定

Python 中的 LarsLasso 算法主要通过 sklearn.linear_model 包实现，其中 LarsLasso 的求解需要设定正则化参数∂，目前主要正则化参数∂为 1、0.1、0.01、0.001 和 0.005。可以通过 LarsLasso_CV 模块可以选择最优正则化参数，最终确定正则化参数$\partial = 0.01$为最优参数。

(2) LarsLasso 算法的变量选择

Lars-LarsLasso 算法主要通过加入回归系数向量的L_1范数）进行惩罚来压缩回归系数的大小，使绝对值较小的回归系数自动被压缩为 0，从而产生稀疏解和实现变量选择，本文数据经过 LARS 算法带入到 LarsLasso 模型中，变量选择结果如图 6-6（回归系数均取绝对值进行展示）。

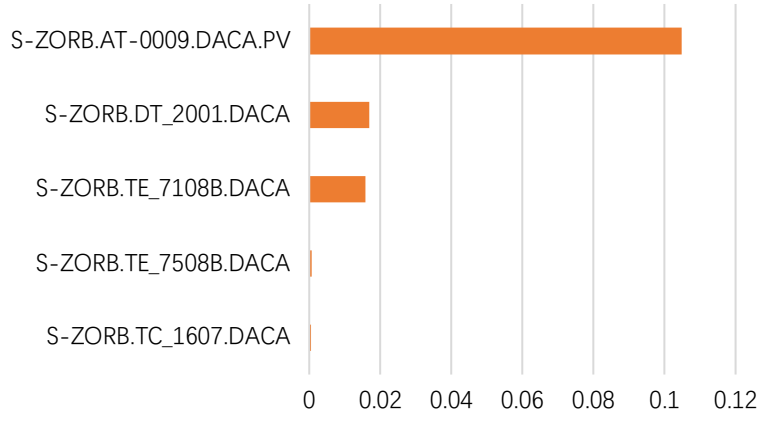


图 6-6 LARSLASSO 特征提取操作变量系数绝对值

如上图所示，选择的变量为 S-ZORB.DT_2001.DACA、S-ZORB.TE_7508B.DACA、S-ZORB.TE_7108B.DACA、S-ZORB.TC_1607.DACA 和 S-ZORB.AT-0009.DACA.PV，未被选择的变量的参数为零。

6.3.4 ElasticNet 算法的求解结果

（1）ElasticNet 算法的参数设定

Python 中的 LarsLasso 算法主要通过 sklearn.linear_model 包实现，其中 LarsLasso 的求解需要设定正则化参数∂，目前主要正则化参数∂为-1.5、-1、-0.5、0、0.5、1 和 1.5。可以通过 ElasticNet_CV 模块可以选择最优正则化参数，最终确定正则化参数$\partial = 0.5$为最优参数。

（2） ElasticNet 算法的变量选择

ElasticNet 算法是在 LarsLasso 算法加入正则化的惩罚范数的基础上，在变量选择上具有以部分变量作为一个整体同时被选中或剔除的特性的自动阻效应稀疏模型和 LarsLasso 算法相似的是，运用惩罚值来压缩不相关的变量，从而产生稀疏解和实现变量选择，本文数据通过 ElasticNet 算法得到的变量选择结果如图 6-7（回归系数均取绝对值进行展示）。

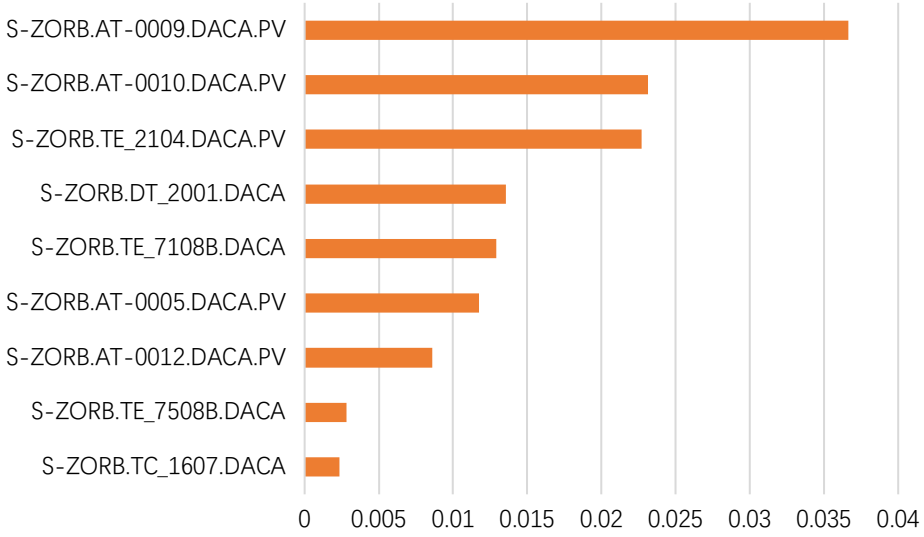


图 6-7 ElasticNet 特征提取操作变量系数绝对值

如上图所示，选择的变量为 S-ZORB.DT_2001.DACA、S-ZORB.TE_7508B.DACA、S-ZORB.TE_7108B.DACA 、 S-ZORB.TC_1607.DACA 、 S-ZORB.AT-0005.DACA.PV 、 S-ZORB.AT-0009.DACA.PV、S-ZORB.AT-0010.DACA.PV、S-ZORB.AT-0012.DACA.PV 和 S-ZORB.TE_2104.DACA.PV，未被选择的变量的参数为零。

6.3.5 RandomForest 算法的求解结果

（1）RandomForest 算法的参数设定

随机森林算法中涉及到较多参数的设定，根据已有的文献中的实证案例将其中的部分重要参数进行设置，n_estimators=10 森林中树的个数。利用 gini impurity（基尼不纯度）度量每种类型的比例，分列一个内部节点所需的最少样本数为 2 个，在叶子节点中需要输入数据的最少加权分数为 0，数增长过程中的最大叶子节点数为正无穷。

（2）RandomForest 算法的变量选择

RandomForest 算法是一种有监督的机器学习算法，被广泛应用于分类与回归问题，它的运算效率高且能够处理高维 （即特征变量多）数据，且具有较好的鲁棒性，无需特征筛选也能得到较高的准确率。本文选择该算法对高维的操作变量进行筛选，希望得到尽可能独立的、具有代表性的主要操作变量，变量选择的结果如图 6-8（回归系数均取绝对值进行展示）。

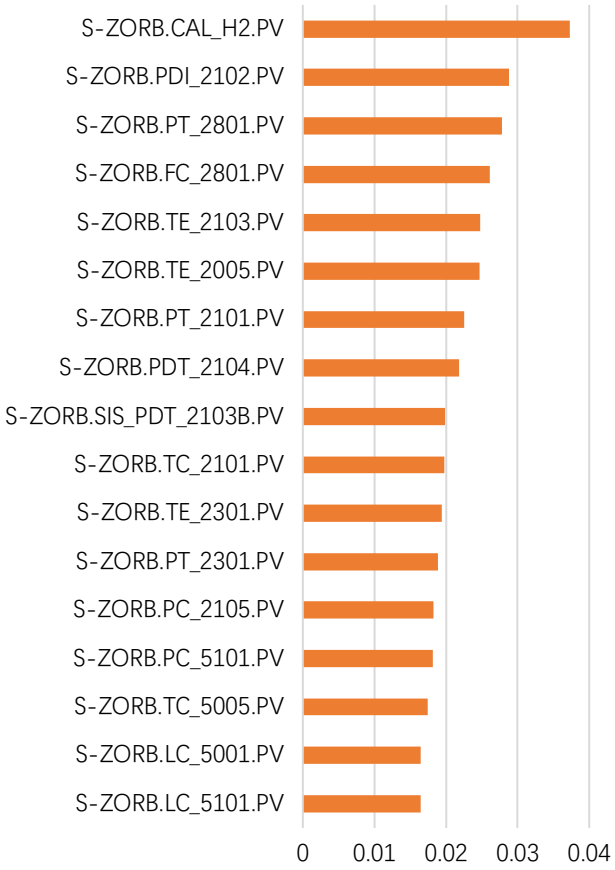


图 6-8 RandomForest 特征提取操作变量系数绝对值

如上图所示，选择的变量为 S-ZORB.CAL_H2.PV、S-ZORB.PDI_2102.PV、S-ZORB.PT_2801.PV、S-ZORB.FC_2801.PV、S-ZORB.TE_2103.PV、S-ZORB.TE_2005.PV、S-ZORB.PT_2101.PV、S-ZORB.PDT_2104.PV、S-ZORB.SIS_PDT_2103B.PV、S-ZORB.TC_2101.PV、S-ZORB.TE_2301.PV、S-ZORB.PT_2301.PV、S-ZORB.PC_2105.PV、S-ZORB.PC_5101.PV、S-ZORB.TC_5005.PV、S-ZORB.LC_5001.PV、S-ZORB.LC_5101.PV 和 S-ZORB.TE_5102.PV，未被选择的变量的参数为零。

6.4 优化后的变量选择

由于精制过程的复杂性和操作设备的多样性，操作变量数量较多，且操作变量之间具有非线性和强耦联的关系，本文采用了五种方法（算法），即逐步回归法、WEIGHT-PCA 法、LarsLasso 法，ElasticNet 法和 RandomForest 法，对变量进行筛选，从而达到降维的作用。最终通过对比和分析五种方法（算法）确定的主要操作变量集，取交集，最终确定的

主要变量是 S-ZORB.PT_2801.PV、S-ZORB.DT_2001.DACA、S-ZORB.TE_7508B.DACA、S-ZORB.TE_7108B.DACA、S-ZORB.TC_1607.DACA 和 S-ZORB.AT-0009.DACA.PV。

七：问题三模型的建立与求解

7.1 问题描述与分析

由于炼油工艺过程的复杂性以及设备的多样性，汽油精制过程中的操作变量之间具有高度非线性和相互强耦联的关系，并且基于传统的数据关联模型中变量相对较少、机理建模对原料的分析要求较高，对过程优化的响应不及时，效果并不理想的背景。我们利用问题一、问题二所选取出来的样本和建模主要变量，通过数据挖掘技术建立辛烷值（RON）损失预测模型，并进行模型验证。

问题三思路流程图如下

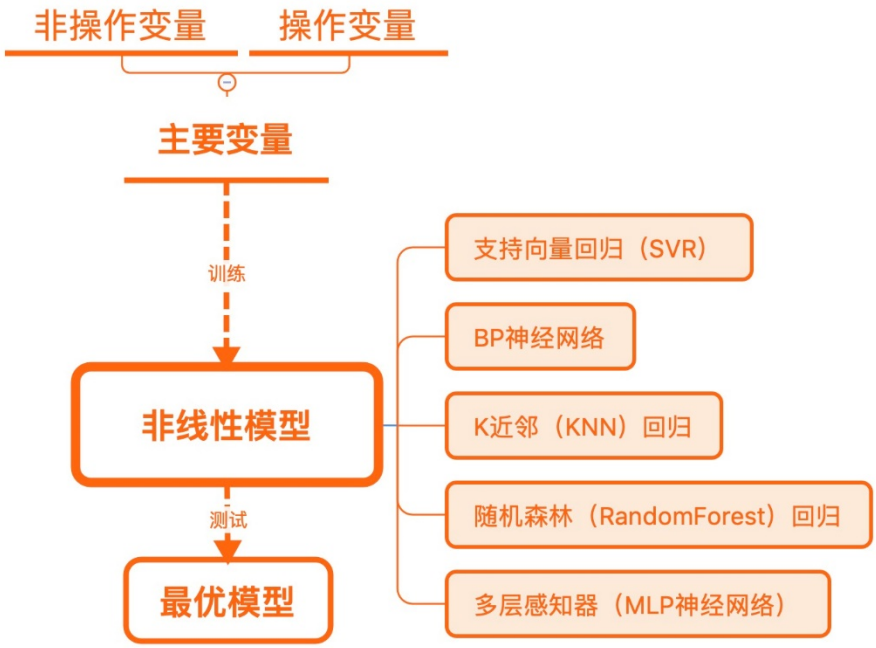


图 7-1 问题三的思路流程图

本文对模型选取的分析如下：

1）基于数据特征较多且特征之间呈现高度非线性和相互强耦联，本文侧重于采用区别与传统回归模型的机器学习非线性回归模型，以提高变量之间关系的解释性、模型的拟合性和预测值的准确性。

2）结合变量特征数据较为稀疏、类域的交叉或重叠较多的样本较多、样本值波动较大且大多数特征的取值都为 0 的数据特点。本文选取出了支持向量回归（SVR）、BP 神经网络、K 近邻（KNN）回归、随机森林回归和多层感知器（MLP 神经网络），分别建立辛烷值（RON）预测模型。

3）以问题一、问题二所选取出来的样本和建模主要变量数据的 75%作为训练集、25%作为测试集进行模型的训练和检验。在测试集上， 将真实值与模型预测值在曲线图中进行

比较，并计算各模型的均方误差（MSE）、均方根误差（RMSE）、平均绝对误差（MAE）、进行综合对比，从而判断模型的准确性，并综合曲线图与误差值表选取最终模型。

7.2 选用的数学模型

7.2.1 支持向量回归（SVR）

支持向量回归(support vector regression, SVR)是一种基于统计学习理论的机器学习算法，其基本思想是。将线性不可回归的样本点通过升维实现线性化。在 SVR 中，目标函数是凸函数，这意味着始终可以达到全局最优。引入核函数的 SVR 可将样本从原始空间映射到一个更高维的特征空间，使得样本在这个特征空间内线性可分，隐式地应用了多项式，因此可以很好地拟合非线性趋势。

令$\emptyset(x)$表示x将映射后的特征向量，于是，在特征向量中划分超平面所对应的模型可表示为

$$f(x)=\omega^T\phi(x)+b \tag{20}$$

解决非线性回归问题其实就等同于求解权重ω_i和阈值b的过程，即对式（21）的二次规划问题进行求解：

$$\min[1/2\,\|\omega\|^2+C\sum_{i=1}^{m}(\xi^*+\xi_i)] \tag{21}$$

$$s.t.\begin{cases} y_i-(\omega\bullet\phi(x))-b\le\varepsilon+\xi_i \\ (\omega\bullet\phi(x))+b-y_i\le\varepsilon+\xi^* \\ \xi_i,\xi_*i\ge 0 \end{cases}$$

一般的机器学习模型中，只有当 *f(x)*与 *y* 完全相同时，损失才为零，而支持向量回归假设我们能容忍的 *f(x)*与 *y* 之间最多有ε的偏差，当且仅当 *f(x)*与 *y* 的差别绝对值大于ε时，才计算损失，此时相当于以 *f(x)*为中心，构建一个宽度为 2ε的间隔带，若训练样本落入此间隔带，则认为是被预测正确的。s

SVR 算法主要有如下主要几个特点：

(1)非线性映射是 SVR 方法的理论基础, SVR 利用内积核函数代替向高维空间的非线性映射；

(2)对特征空间划分的最优超平面是 SVR 的目标,最大化分类边际的思想是 SVR 方法的核心；

(3)SVR 的最终决策函数只由少数的支持向量所确定,计算的复杂性取决于支持向量的数目,而不是样本空间的维数,这在某种意义上避免了“维数灾难”。

(4)少数支持向量决定了最终结果,这不但可以帮助我们抓住关键样本、“剔除”大量冗余样本,具有较好的稳健性。

7.2.2 BP 神经网络

BP 神经网络是一种基于误差反向传播的神经网络，具有自组织、自适应和自学习的能力，能够实施大规模的并行处理。同时其具有的非线性映射特性，大大地增强了适应环境的能力，具有较好的鲁棒性和容错性。误差反向传播网络模型利用已知数据通过迭代梯度算法求解网络的实际输出与期望输出之间的最小均方差值，并将信息反向传递和

修改误差。在误差反向传播的过程中不断地对权值和阈值进行修正，以此达到降低误差的目的，使网络对输入模式响应的正确率不断提升。其基本结构如下图所示:

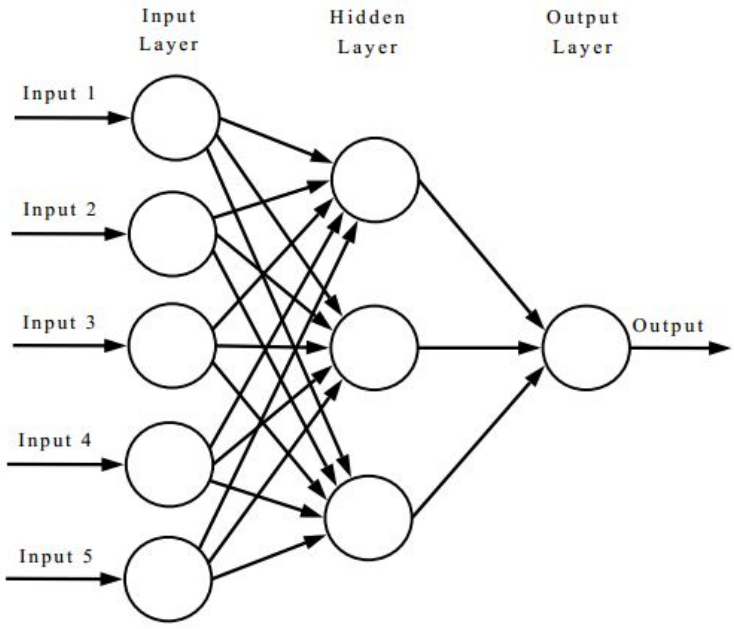


图 7-2 BP 神经网络基本结构

BP 神经网络实质上实现了一个从输入到输出的映射功能，典型的 BP 神经网络一般包括输入层、输出层和隐含层，BP 神经网络在进行学习和训练时主要考虑因素有隐含层数、隐含层神经元数、网络权值、期望误差和学习速率。本网络采用 3 层的神经网络结构，即单隐含层的神经网络。数学理论证明三层的神经网络就能够以任意精度逼近任何非线性连续函数。隐含层神经元个数通过经验公式$M=(n+m)^{1/2}+a$得到，其中，M表示隐含层神经元个数，n和m分别表示输入层和输出层神经元个数a是区间 [0,10] 的常数。根据公式最终求出神经元个数在 4～14 之间，之后通过试算法确定。初始权值采用系统默认的随机数，学习速率的取值范围 [0, 1]，根据经验一般取 0.01～0.8。

7.2.3 KNN 回归

KNN 回归算法是一种基于实例的学习方法，其核心思想是建立向量空间模型，基于某种距离度量方式，找到训练集中与测试点最接近的 k 个近邻点，利用这 k 个近邻点对测试集进行预测，在回归问题中常采用“平均法”，即这 k 个近邻点输出的平均值作为预测结果。KNN 回归算法的具体步骤如下：

（1）将训练集设为$X_i=(x_1^i,x_2^i,\dots,x_3^i,y_n)$，将测试集某点设为$X_i=(x_1,x_2,\dots,x_ny_i)$。

（2）遍历训练集中各点x_i，求其与测试集中某点的欧氏距离L:

$$L(X,X_i)=\sqrt{\sum_{m=1}^{n}(x_m-x_i^m)^2} \tag{22}$$

（3）对求得的距离大小排序，选择训练集与点X最近的 k 个近邻点$X_i(1\le i\le k)$，这 k 个近邻点的输出的平均值作为X的输出预测值，即：

$$\hat{y} = \sum_{j=1}^{k} y_j / k \tag{23}$$

其中，$\hat{y}$为X的输出预测值。

一般情况下，KNN 算法有 2 个重要参数：邻居个数与数据点之间距离的度量方法。本文选取的邻近个数 k 为 6，距离度量方式采用的是欧式距离。

KNN 回归模型的理论成熟，思想简单，用来做非线性回归具有一定的优势。其训练时间复杂度低，仅为 O(n)，和朴素贝叶斯之类的算法比，对数据没有假设，准确度高，对异常点不敏感。由于 KNN 方法主要靠周围有限的邻近的样本，而不是靠判别类域的方法来确定所属类别的，因此对于类域的交叉或重叠较多的待分样本集来说，KNN 方法较其他方法更为适合。

7.2.4 随机森林

随机森林算法是一种较新的高度灵活的一种机器学习算法。该算法的基本单元是决策树，但其本质属于机器学习中的方法。

随机森林算法主要过程如下：

（1）样本集的选择

假设原始样本集总共有 N 个样例，则每轮从原始样本集中通过 Bootstraping（有放回抽样）的方式抽取 N 个样例，得到一个大小为 N 的训练集。在原始样本集的抽取过程中，可能有被重复抽取的样例，也可能有一次都没有被抽到的样例。共进行 k 轮的抽取，则每轮抽取的训练集分别为 T1,T2,…,Tk。

（2）决策树的生成

假如特征空间共有 D 个特征，则在每一轮生成决策树的过程中，从 D 个特征中随机选择其中的 d 个特征（d<D）组成一个新的特征集，通过使用新的特征集来生成决策树。在 k 轮中共生成 k 个决策树，由于这 k 个决策树在训练集的选择和特征的选择上都是随机的，因为这 k 个决策树之间是相互独立的。

（3）模型的组合

由于生成的 k 个决策树之间是相互独立的，每个决策树的重要性是相等的，因而在将它们进行组合时，无需考虑它们的权值，或者可以认为它们具有相同的权值。对于分类问题，最终的分类结果使用所有的决策树投票来确定最终分类结果；对于回归问题，使用所有决策时输出的均值来作为最终的输出结果。具体的流程图如下：

（4）模型的验证

模型的验证需要验证集，但不需要额外专门再获取验证集，只需要从原始样本集中选择没有被使用过的样例即可。

在从原始样本中选择训练集时，存在部分样例一次都没有被选中过，在进行特征选择时，也可能存在部分特征未被使用的情况，我们只需将这些未被使用的数据拿来验证最终的模型即可。

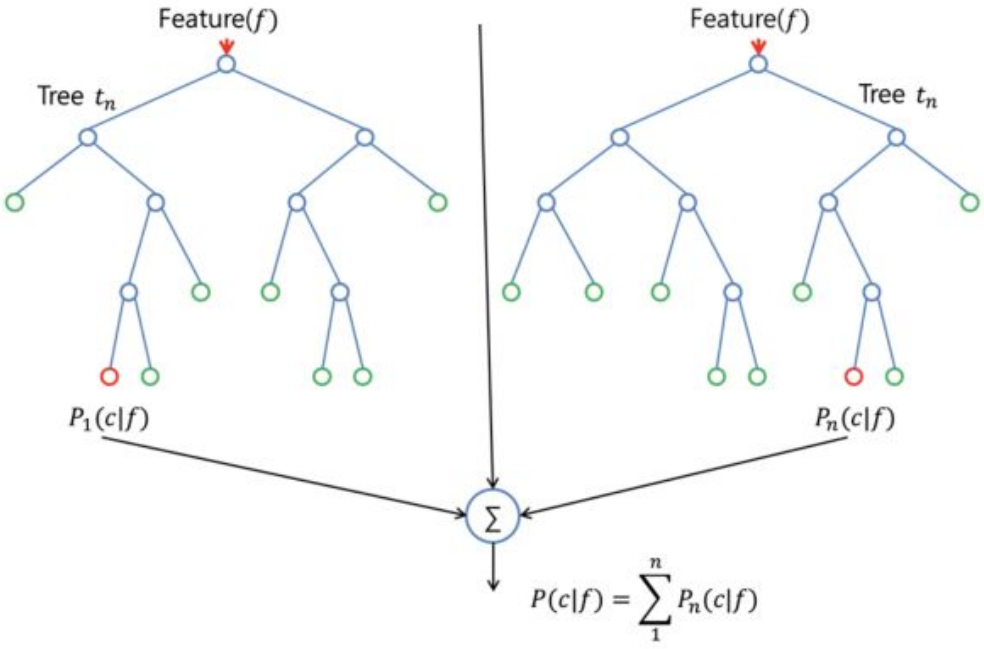


图 7-3 随机森林图解

随机森林算法主要有以下 2 个特性：

（1）有两个随机抽取过程：分别为从原始样本集中随机抽取训练集，和决策树的特征选择上随机抽取部分特征生成决策树。

（2）各决策树之间的相互独立性。因为相互独立，故而在决策树的生成过程可以并行进行，大大提高了算法的时间效率。

7.2.5 多层感知器

多层感知器(Multi-Layer Perceptron, MLP)是一种人工神经网络结构。由多个感知器单元组成的神经网络，如图 X 所示。每一层的所有感知器与下一层的所有感知器相连。MLP 由输入层、输出层和一系列中间层(隐藏层)组成。每一层由一个或多个感知器单元组成。MLP 的权重值和偏差值是随机梯度下降过程中 更新的可训练参数。综上所述，可以将 MLP 表示为

$$F(x) = s \circ \lambda_n \circ \sigma_{n-1} \circ \lambda_{n-1} \circ \cdots \circ \lambda_1(x) = y \tag{24}$$

其中 n 是全连接层，f 为激活函数，即 Softmax 函数。

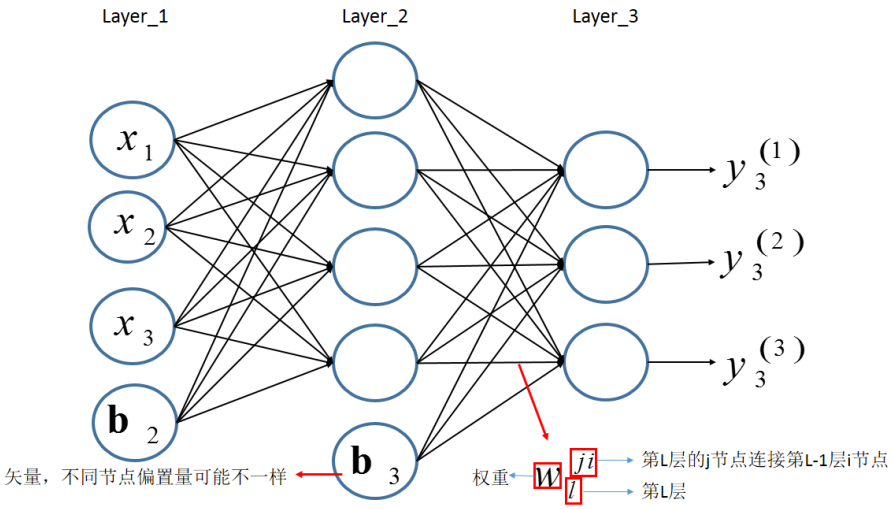


图 7-4 MLP 网络结构

MLP 神经网络是由神经元形成一个网格状的结构，该结构被分成多个连接层，每个神经元的值为

$$F(x)=s\circ\lambda_n\circ\sigma_{n-1}\circ\lambda_{n-1}\circ\cdots\circ\lambda_1(x)=y \tag{25}$$

其中ω为相邻层之间神经元连接权重值，b 为该神经元的偏置值，f 为激活函数，当前层的神经元都是上一层中每个相连神经元的输出值的函数。

MLP 网络模型算法核心思想是通过前向传播得到误差，再把误差通过反向传播实现权重值ω的修正，最终得到最优模型。在反向传播过程中通常使用随机梯度下降法对权重值进行修正，梯度下降法的原理是计算损失函数关于所有内部变量的梯度，并进行反向传播。内部变量通常是权重值，根据损失函数所跨越曲面的最陡下降方向进行调整。

7.3 模型的构建与评价

本章对于炼油工艺过程中影响辛烷值（RON）损失值的主要因素之间的关联性进行了分析，并搜构建出了四种辛烷值（RON）损失预测模型。下面对五种模型进行比较分析，并根据预测图像与误差体系进行模型的评价与选取。

利用 Python 构建五种模型，其中 SVR 采用了线性核、多项式核、径向基核三种内核；BP 神经网络采用三层网络，两层隐藏层神经单元数分别为 14、11；KNN 回归采用 n_neighbors 为 2 的结构进行拟合；随机森林回归采用 n_estimator=10 的结构进行拟合；MLP 神经网络采用三层网络结构进行拟合。

模型名称	参数设置
SVR	线性核、多项式核、径向基核
BP 神经网络	三层网络，两层隐藏层神经单元数分别 14、11
KNN 回归	n_neighbors=2
随机森林回归	n_estimator=10
MLP 神经网络	三层网络

表 7-1 五种模型基本参数

将数据集的 75%作为训练集，25%作为测试集，各个模型在测试集上的预测效果如下图所示

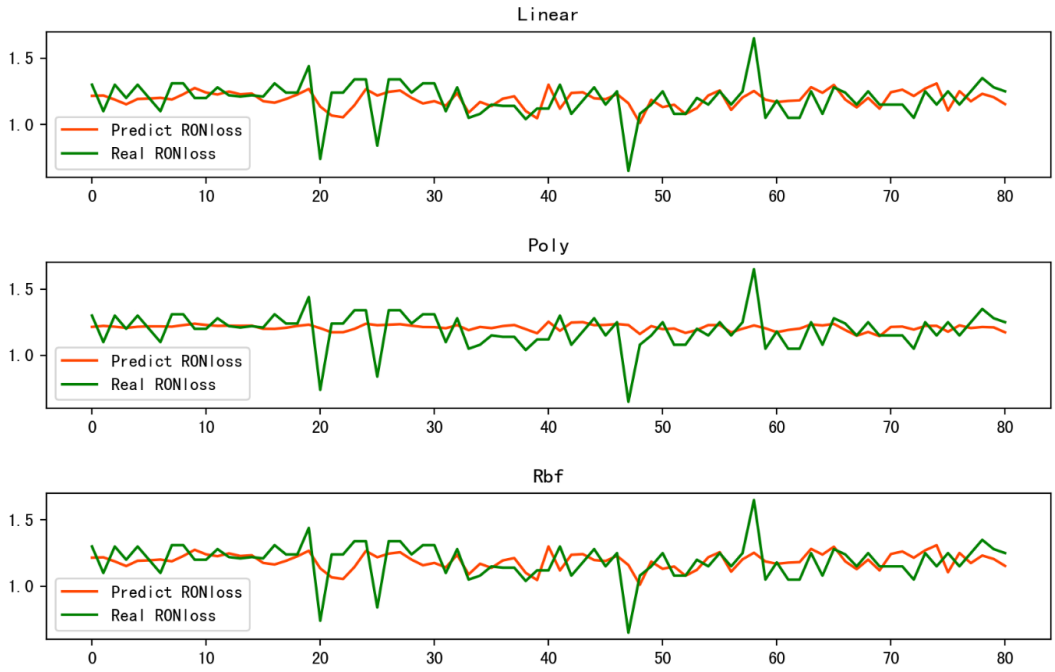


图 7-5 三种核函数 SVR 模型拟合情况

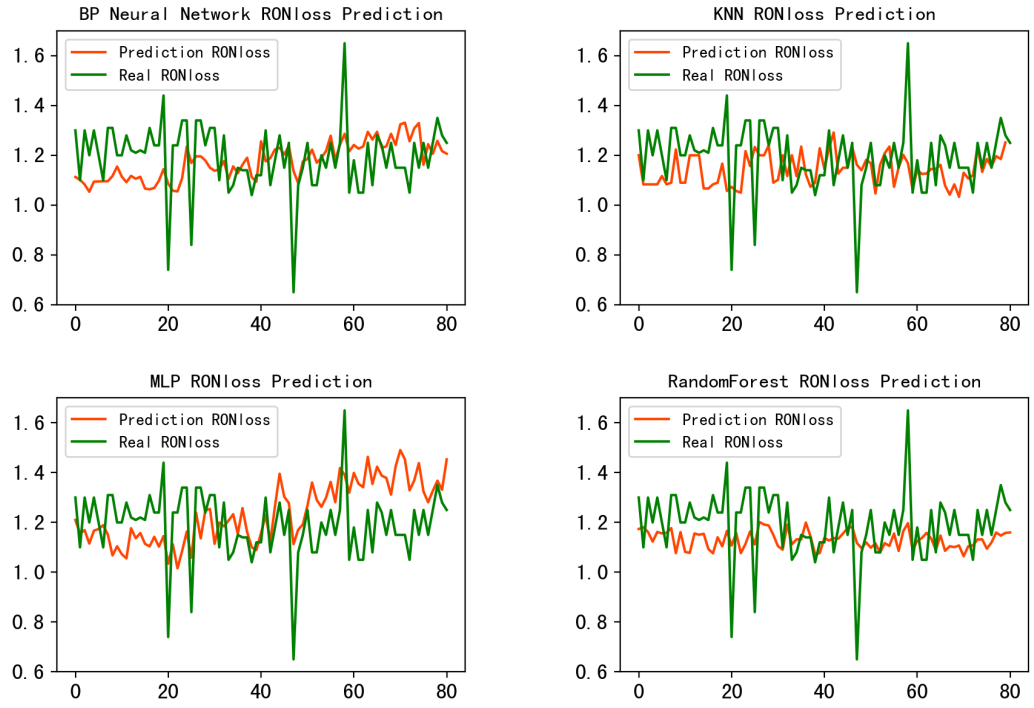


图 7-6 MLP 算法拟合情况

在构建好模型之后，本文进行了误差分析，用于验证和比较各模型的预测能力和准确性。因此，经过计算，本文将构建均方误差（MSE）、均方根误差（RMSE）、平均绝对误差（MAE）的综合误差指标体系进行对比。如表 7-2 所示

模型	MSE	RMSE	MAE
SVR 线性核	0.023475	0.153216	0.118108
SVR 多项式核	0.641367	0.800854	0.632684
SVR 径向基核	0.025857	0.160802	0.120929
BP 神经网络	0.018565	0.136255	0.102374
K 近邻（KNN）回归	0.023589	0.153586	0.118104
随机森林回归	0.019814	0.140761	0.108544
多层感知器(MLP 神经网络)	0.024732	0.157265	0.130882

表 7-2 多种模型的误差值表情况

根据上述五个模型在测试集上的拟合曲线以及误差情况综合分析，我们选取了 SVR 模型。理由如下：

（1）本文研究的主要变量为在炼油工艺过程中所涉及到的多个操作变量，变量特征呈现高度复杂的非线性和相互的强耦联性。而 SVR 模型在解决高维特征的分类问题和回归问题时呈现出独特的优越性,即使在特征维度大于样本数时依然有很好的效果，因此该模型可以很好解决本文数据高维特征的问题。

（2）由于 SVR 模型仅仅使用一部分支持向量来做超平面的决策，无需依赖全部数据，因此数据的增删对模型效果没有影响。此特征适用于本文数据来自于前两题对样本数据的主要特征的提取和处理。

（3）SVR 有多个核函数可以选择，在引入核函数后，模型可以很灵活地解决各种复杂的非线性回归问题，模型的解释性很强。

（4）在样本量不是海量数据的时候，预测准确率高，泛化能力强。文本主要采取 325 个变量的 18 个主要特征来作为模型的输入，输入数据量属于中等，不算大。符合该特征。

综上，SVR 的特征与本文数据的特征高度吻合，因此该模型在本节的模型选择中最为合适，该模型具有很好的预测能力,且其预测模型推广能力也很强,即使在样本值波动很大的情况下预测仍然具有较高的精度、算法收敛速度快且稳定，因此本章中采用模型 2 作为本题的主要模型。

八：问题四的模型建立与求解

8.1 问题描述与分析

问题四要求根据本文处理后的数据和筛选后的主要操作变量建立科学合理的辛烷值损失预测模型，在保持非操作变量不变的情况下，分析辛烷值损失下降对应的主要操作变量的变化程度，从而指导精制过程中减少辛烷值损失的操作方案，为精制过程的优化提供理论估计范围。问题二中已经利用五个方法（算法）（即逐步回归法、WEIGHT-PCA 法、LarsLasso 算法，ElasticNet 算法和 RandomForest 算法）筛选出主要操作变量。最终，我们保留了 12 个非操作变量、6 个操作变量。根据附件四中的操作变量范围与Δ值，我们可以构造这 6 个操作变量可能产生的所有动作（操作变量数值组合），进而利用问题三中构造的辛烷值（RON）损失预测模型进行回归，从中选择 RON 损失值最小的动作向量，使得

在保证产品硫含量不大于 5μg/g 的前提下，尽可能的将辛烷值（RON）损失降低。具体的问题解答思路如图 8-1。

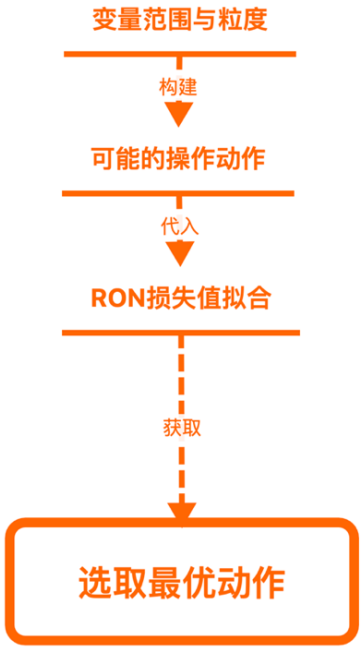


图 8-1 问题四的解答思路流程图

通过如上流程图可以发现，问题四是在已有的辛烷值（RON）损失预测模型的基础上，不断优化操作变量，从而在保证精制后的汽油符合环保属性后尽可能的降低辛烷值损失降，从而为精制过程的操作优化提供科学性的指导意见。

8.2 操作动作的构造和最优值的选取

8.2.1 操作动作的构造

根据题目要求，我们将所有样本的产品硫含量都设置为 5μg/g。由于问题二中已通过合理的模型从 354 个操作变量中筛选出 6 个最具代表性的主要操作变量，根据附件四，我们获取了这 6 个主要操作变量的范围与Δ值，据此构造出 150321600 个可能的操作动作（即主要操作变量的组合）。由于部分操作变量的动作较多（如 S-ZORB.DT_2001.DACA、S-ZORB.TE_7108B.DACA 等），会使模型探索的时间成本过高，因此，对于这些操作变量，我们按照$N_i \times \Delta$的粒度进行动作选取。根据操作变量对应的变化范围得到如下操作动作参数表：

操作变量名称	操作 最小值	操作 最大值	Δ	N	动作 个数	样本 最小值	样本 最大值
S-ZORB. PT_2801.PV	2.35	2.7	0.1	1	4	2.3860	2.6078
S-ZORB. DT_2001.DACA	0	120	10	1	13	0.3017	100.8192

S-ZORB.TE_7508B.DACA	2	100	1	10	10	2.8081	83.2226
S-ZORB.TE_7108B.DACA	3	75	1	4	19	3.6844	64.3965
S-ZORB.TC_1607.DACA	320	480	2	10	9	334.9940	457.8239
S-ZORB.AT-0009.DACA.PV	0.4	0.8	0.1	1	5	0.4305	0.6883

表 8-1 操作动作参数表

通过对分操作变量粒度的调整和处理，最终得到 444600 个实际动作。

8.2.2 最优值的选取

将构造的所有实际动作带入问题三中的辛烷值（RON）预测模型中进行回归分析，选择平均 RON 损失值减少最大的操作变量数值组合（最优动作），结果如下表所示：

操作变量名称	实际操作值
S-ZORB.PT_2801.PV	2.65
S-ZORB.DT_2001.DACA	0
S-ZORB.TE_7508B.DACA	92
S-ZORB.TE_7108B.DACA	3
S-ZORB.TC_1607.DACA	480
S-ZORB.AT-0009.DACA.PV	0.8

表 8-2 最优动作

8.2.3 最优动作的辛烷值（RON）损失预测

经过如上最优动作，325 个样本的平均 RON 损失减少量为 46.49%，远超过题目的要求，说明模型和最优动作的建立的效果较好。最优动作的辛烷值（RON）损失预测如下图 8-1。

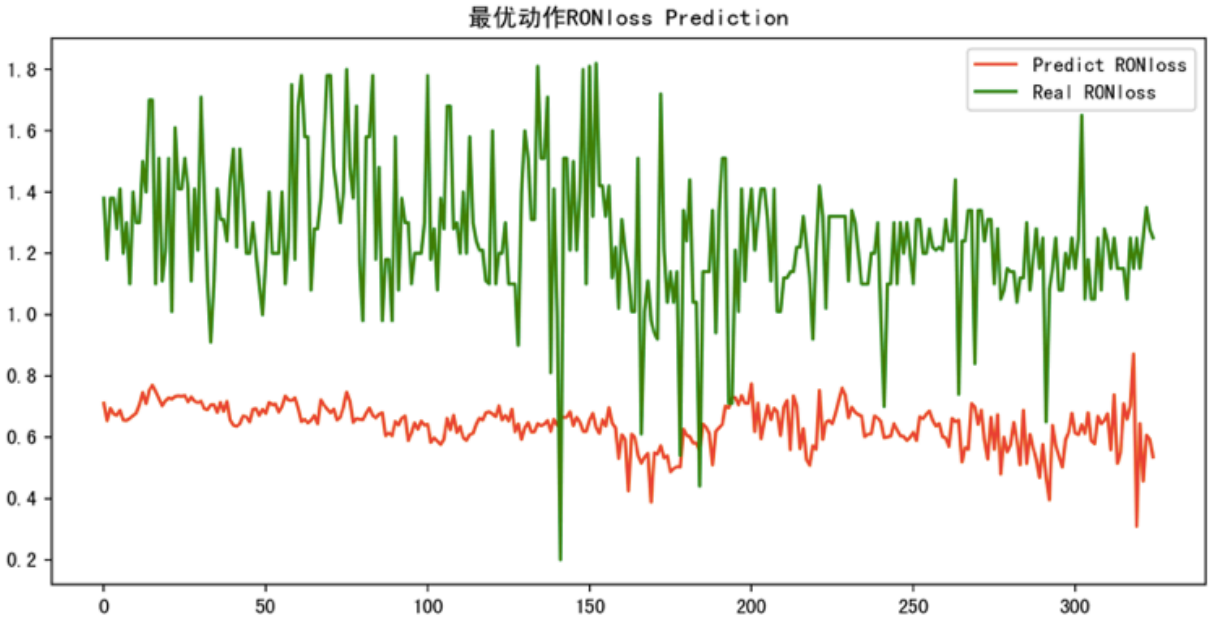


图 8-1 最优动作的辛烷值（RON）损失预测

通过上图，我们发现经过最优操作后 316 个样本的辛烷值损失有下降，且降幅符合题目要求，能够为为精制过程的操作优化提供科学性的指导意见，具有较好的实际意义。

九：问题五的模型建立与求解

9.1 问题描述与分析

由于汽油还需要符合国家的环保要求，在精制过程中，硫含量也是一个非常值得关注的变量，问题五要求我们清晰和直观的展示在操作变量优化调整过程中辛烷值和硫含量的变化轨迹，使汽油精制过程后的汽油满足质量和环保的双重要求。具体的解决思路如图 9-1 所示。

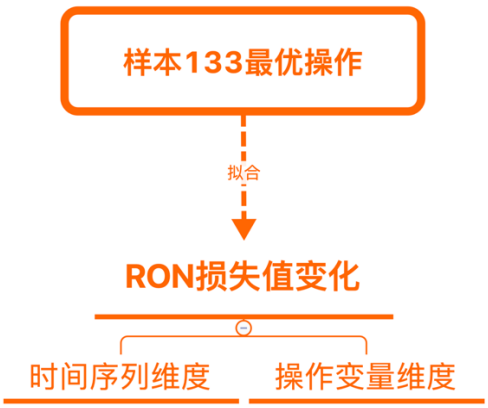


图 9-1 问题五的解题思路

对于非操作变量中的产品硫含量，我们根据时间序列逐渐将其上升到 5μg/g 的水平；对于操作变量，我们获取了样本 133 的 6 项操作变量原始值，利用问题四获得的最优动作以及附件四中的Δ值计算中间动作。

对于优化过程，我们考虑了时间维度与操作变量维度上的分析。（1）在时间序列上，我们假设每一个单位时间段，6 个操作变量可以同时变换对应的Δ值，可以得到整个动作变化过程每一个时间点上的中间动作，并分析该时间点上的 RON 损失值，由此分析 RON 损失值与时间序列之间的关系；（2）在操作变量上，我们选取了需要调整时间较长的三个操作变量，并分析在这三个操作变量变化的过程中，RON 损失值变化情况。

9.2 优化过程分析

9.2.1 133 号样本数据的优化结果

首先，我们需要对 133 号样本的原始数据和优化数据进行对比和分析，具体如下表：

类别		原始动作	最优动作
非操作变量	硫含量,μg/g	248	248
	辛烷值 RON	89.4	89.4
	饱和烃,v%（烷烃+环烷烃）	55.9	55.9
	烯烃,v%	20.6	20.6
	芳烃,v%	23.5	23.5
	溴值,gBr/100g	50.11	50.11
	密度(20℃), kg/m³	727.8	727.8
	硫含量,μg/g.1	3.2	5
	焦炭,wt%	2.53	2.53
	S, wt%	8.57	8.57
	焦炭,wt%.1	1.3	1.3
	S, wt%.1	6.69	6.69
操作变量	S-ZORB.PT_2801.PV	2.4769	2.65
	S-ZORB. DT_2001.DACA	70.6343	0
	S-ZORB.TE_7508B.DACA	4.2648	92
	S-ZORB.TE_7108B.DACA	49.1861	3
	S-ZORB.TC_1607.DACA	406.7505	480
	S-ZORB.AT-0009.DACA.PV	0.4738	0.8
	RON 损失值	1.31	0.88
优化效果	32.80%		

表 9-1 133 号样本的原始数据和优化数据的对比

根据上表的优化效果可以发现，通过优化后的 RON 损失值下降了 32.8%，优化效果较好。

9.2.2 133 号样本数据时间维度的优化过程

假设每一单位时间，六个操作变量可以同时进行一次Δ变化，那么根据样本 133 的原始操作变量与优化后的实际操作变量之间的差距，可以得到需要经过 87 个单位时间，操作变量能够调整到优化后的动作上，每个主要操作变量的操作时间如下表所示：

操作变量名称	原始变量数值	优化后的数值	Time-steps
S-ZORB.PT_2801.PV	2.4769	2.65	2
S-ZORB.DT_2001.DACA	70.6343	0	7
S-ZORB.TE_7508B.DACA	4.2648	92	87
S-ZORB.TE_7108B.DACA	49.1861	3	46
S-ZORB.TC_1607.DACA	406.7505	480	37
S-ZORB.AT-0009.DACA.PV	0.4738	0.8	3

表 9-2 主要操作变量的操作时间

同时我们根据产品的环保要求（即硫含量的要求），设置产品硫含量从原始的 3.2μg/g 逐步提高到 5μg/g，使得 RON 损失率在满足硫含量要求的前提下达到最小，产品硫含量随时间的变化和辛烷值（RON）损失值随时间的变化分别如图 9-2、图 9-3 所示。

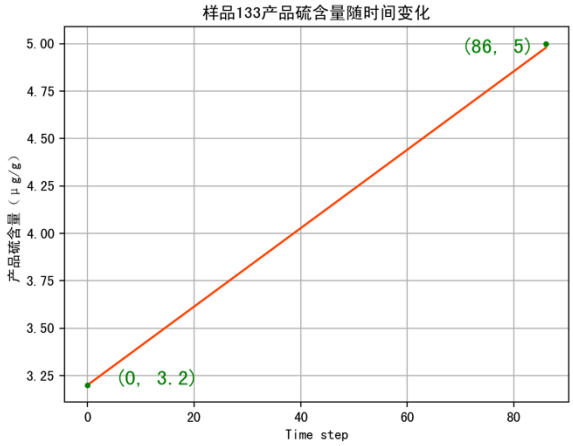


图 9-2 产品硫含量随时间的变化

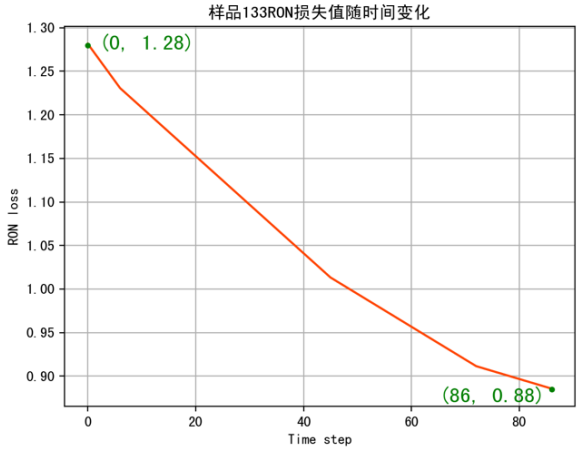
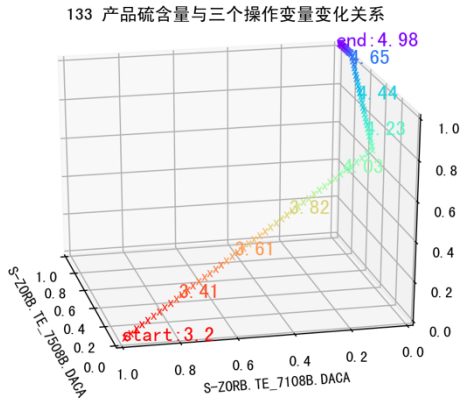


图 9-3 辛烷值损失随时间的变化

9.2.3 133 号样本数据操作变量维度的优化过程

根据原始操作变量与优化后的操作变量之间的差距，我们可以看到 S-ZORB.TE_7508B.DACA、ZORB.TE_7108B.DACA 和 S-ZORB.TC_1607.DACA 的操作过程

最为复杂，下图 8-4 和图 8-5 为这三个操作变量在变动的过程中，硫含量、RON 损失值的变化情况。（为保证图片美观，三个操作变量已进行标准化）



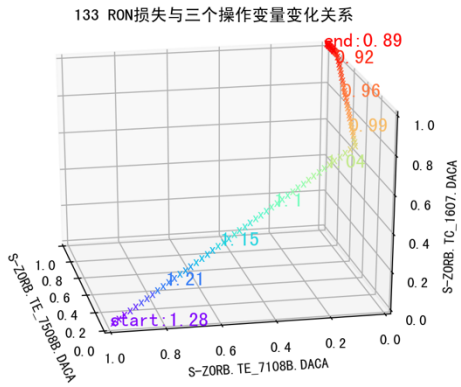


图 9-4 产品硫含量随操作变量变动的变化　图 9-5 辛烷值损失随操作变量变动的变化

十：模型评价

10.1 模型的优点

本文针对不同问题，综合运用了多种统计方法和模型，并对不同模型的结果进行对比和分析，从而得到最优结果。

（1）针对原始数据空缺值，本文独创性的运用反距离权重插值方法对空值进行科学合理的补充，有效的解决了出现异常值和空缺值的样本问题；

（2）特征提取过程中采用了逐步回归法、WEIGHT-PCA法、LarsLarsLasso算法、ElasticNet算法、RandomForest算法共五种模型，综合考虑多种方法提取出的变量特征，准确的提取出了高维样本中的关键特征信息，对原始数据中的高维参数进行合理降维；

（3）回归分析中我们采用SVR模型对辛烷值（RON）损失预测模型进行构建，同时用BP神经网络回归、K临近（KNN）回归、RandomForest回归和多层感知器（MLP神经网络）等多种算法对模型进行验证；

（4）最优操作的获取中，基于操作变量构造的动作可以保证在较高精度情况下获取最优的操作变量组合。

综上，在各个关键问题中，我们几乎对所有模型都进行了检验和对比分析。我们在对各个问题进行数据处理、模型搭建与优化求解的过程中，总和运用了SPSS、Excel、Python等统计分析软件对数据进行分析处理，使得各项数据、模型真实可信，同时利用多种图表展示分析过程与结果，使得整个问题的分析和求解过程清晰直观。

10.2 模型的缺点

（1）由于附件 1 中操作变量高达三百多个，且初始样本仅有三百余条，给特征选取带来了较大的困难，即使在多个特征提取方法综合考虑的情况下，仍会出现遗漏部分可能存在的重要变量信息；

（2）在模型构建上，得到的拟合效果仍有提高的空间，也会对后面利用模型优化操作变量造成一定程度的影响。

10.3 模型的展望

（1）我们可以利用其他非线性特征提取方法对操作变量进行进一步分析，从而确保重要变量信息得以保留；

（2）在模型构建上，我们需要利用一些专家知识和相关的基础理论更好的选择模型，例如一些与时间序列高度相关的操作变量使用基于循环神经网络优化的长短记忆神经网络（LSTM）可能会有更好的拟合效果。

十一：参考文献

[1].Becker N, Toedt G, Lichter P, et al. Elastic SCAD as a novel penalization method for SVM classification tasks in high-dimensional data[J]. BMC bioinformatics, 2011, 12(1): 138.

[2].Bijlsma S, Bobeldijk I, Verheij E R, et al. Large-scale human metabolomics studies: a strategy for data (pre-) processing and validation[J]. Analytical chemistry, 2006, 78(2): 567-574.

[3].Drucker H, Burges C J C, Kaufman L, et al. Support vector regression machines[C]//Advances in neural information processing systems. 1997: 155-161.

[4].Fu W J. Penalized regressions: the bridge versus the LarsLasso[J]. Journal of computational and graphical statistics, 1998, 7(3): 397-416.

[5].Gao J, Kwan P W, Shi D. Sparse kernel learning with LARSLASSO and Bayesian inference algorithm[J]. Neural Networks, 2010, 23(2): 257-264.

[6].Hecht-Nielsen R. Theory of the backpropagation neural network[M]//Neural networks for perception. Academic Press, 1992: 65-93.

[7].Huang J, Horowitz J L, Ma S. Asymptotic properties of bridge estimators in sparse high-dimensional regression models[J]. The Annals of Statistics, 2008, 36(2): 587-613.

[8].Keller J M, Gray M R, Givens J A. A fuzzy k-nearest neighbor algorithm[J]. IEEE transactions on systems, man, and cybernetics, 1985 (4): 580-585.

[9].Liaw A, Wiener M. Classification and regression by randomForest[J]. R news, 2002, 2(3): 18-22.

[10].Ravikumar P, Wainwright M J, Lafferty J D. High-dimensional Ising model selection using ℓ1-regularized logistic regression[J]. The Annals of Statistics, 2010, 38(3): 1287-1319.

[11].Ruck D W, Rogers S K, Kabrisky M, et al. The multilayer perceptron as an approximation to a Bayes optimal discriminant function[J]. IEEE Transactions on Neural Networks, 1990, 1(4): 296-298.

[12].Tibshirani R. Regression shrinkage and selection via the LarsLasso: a retrospective[J]. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 2011, 73(3): 273-282.
[13].Tibshirani R. Regression shrinkage and selection via the LarsLasso[J]. Journal of the Royal Statistical Society: Series B (Methodological), 1996, 58(1): 267-288.
[14].Wold S, Esbensen K, Geladi P. Principal component analysis[J]. Chemometrics and intelligent laboratory systems, 1987, 2(1-3): 37-52.
[15].Yuan M, Lin Y. Model selection and estimation in regression with grouped variables[J]. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 2006, 68(1): 49-67.
[16].Zhang C H, Huang J. The sparsity and bias of the LarsLasso selection in high-dimensional linear regression[J]. The Annals of Statistics, 2008, 36(4): 1567-1594.
[17].Zhang C, Wang J, Zhao N, et al. Reconstruction and analysis of multi-pose face images based on nonlinear dimensionality reduction[J]. Pattern Recognition, 2004, 37(2): 325-336.
[18].白琳, 徐永明, 何苗, 等. 基于随机森林算法的近地表气温遥感反演研究[J]. 地球信息科学学报, 2017, 19(3): 390-397.
[19].刘建伟, 崔立鹏, 刘泽宇, 等. 正则化稀疏模型[J]. 计算机学报, 2015, 38(7): 1307-1325.
[20].刘潇, 薛莹, 纪毓鹏, 等. 基于主成分分析法的黄河口及其邻近水域水质评价[J]. 中国环境科学, 2015, 35(10): 3187-3192.
[21].尹峻松, 肖健, 周宗潭, 等. 非线性流形学习方法的分析与应用[J]. 自然科学进展, 2007, 17(8): 1015-1025.
[22].游士兵, 严研. 逐步回归分析法及其应用[J]. 统计与决策, 2017 (14): 31-35.
[23].约翰逊. 实用多元统计分析: 英文本[M]. 清华大学出版社, 2008.

十二：附录

问题一的 Python 代码

```
# -*- coding: utf-8 -*-
"""
Created on Thu Sep 17 10:53:26 2020
@author: al
"""
import pandas as pd
import re
import numpy as np
from collections import defaultdict

#1-1
#根据附件四的变量范围，将超出范围的数据改为*，并存在 285t/313t.xlsx 中
#如果附件四中的变量范围没有跨越 0，则认为 0 数据是空值，0<=min or 0 >=max
df285 = pd.read_excel('285.xlsx')
df313 = pd.read_excel('313.xlsx')
```

```
columns = list(df285.columns)[1:]
R = pd.read_excel('range.xlsx')
#print(list(R['range']))
R = list(R['range'])
for i in range(len(R)):
    pattern = re.compile('(.*)-(.*)')
    t = pattern.findall(R[i])
    #print(i, end='\t')
    #print(t, end='\t')
    try:
        min_r = float(t[0][0])
        max_r = t[0][1]
        if max_r.startswith('(') or max_r.startswith('（'):
            max_r = float(max_r[1:-1])
        else:
            max_r = float(max_r)
    except:
        min_r = -(float(t[0][0].split('-')[1]))
        max_r = -(float(t[0][1][:-1]))
        #print('**', end='\t')
    for j in range(40):
        if float(df285.iloc[j, i+1]) <= min_r or float(df285.iloc[j, i+1]) >= max_r:
            df285.iloc[j, i+1] = np.nan
        if float(df313.iloc[j, i+1]) <= min_r or float(df313.iloc[j, i+1]) >= max_r:
            df313.iloc[j, i+1] = np.nan
df285.to_excel('285a.xlsx')
df313.to_excel('313a.xlsx')

#1-2
#根据 2/8 法则，获取数据量少于 80%的 columns
failist285 = []
for i in range(len(columns)):
    n = 0
    for j in range(40):
        if str(df285.iloc[j, i+1]) == 'nan':
            n += 1
    if n > 8:
        failist285.append(columns[i])
print(failist285)
failist313 = []
for i in range(len(columns)):
    n = 0
    for j in range(40):
        if str(df313.iloc[j, i+1]) == 'nan':
```

```
                n += 1
        if n > 8:
            failist313.append(columns[i])
print(failist313)

#1-3
#线性插值
df285 = df285.interpolate(limit_direction='both')
df313 = df313.interpolate(limit_direction='both')
df285.to_excel('285b.xlsx')
df313.to_excel('313b.xlsx')

#1-4
#保存 2/8 法则留存下来以及线性插值后的数据并求取平均值
c285 = defaultdict(float)
for i in range(len(columns)):
    s = 0.0
    n = 0
    if not columns[i] in failist285:
        for j in range(40):
            if str(df285.iloc[j, i+1]) != 'nan':
                n += 1
                s += float(df285.iloc[j, i+1])
        c285[columns[i]] = s/n
    else:
        c285[columns[i]] = 0
#print(dict(c285))
df285c = pd.DataFrame.from_dict(dict(c285),orient='index').T
df285c.to_excel('285c.xlsx')
c313 = defaultdict(float)
for i in range(len(columns)):
    s = 0.0
    n = 0
    if not columns[i] in failist313:
        for j in range(40):
            if str(df313.iloc[j, i+1]) != 'nan':
                n += 1
                s += float(df313.iloc[j, i+1])
        c313[columns[i]] = s/n
    else:
        c313[columns[i]] = 0
#print(dict(c313))
df313c = pd.DataFrame.from_dict(dict(c313),orient='index').T
df313c.to_excel('313c.xlsx')
```

第二部分

```
# -*- coding: utf-8 -*-
"""
Created on Thu Sep 17 15:02:05 2020
@author: al
"""
import pandas as pd
import re
import numpy as np
from collections import defaultdict
df = pd.read_excel('样本数据.xlsx')
length = df.shape[0]
#print(df.head())

#2-1
#根据附件四范围删除异常数据
#判断 0 值是否为空
R = pd.read_excel('range.xlsx')
#print(list(R['range']))
R = list(R['range'])
for i in range(len(R)):
    pattern = re.compile('(.*)-(.*)')
    t = pattern.findall(R[i])
    #print(i, end='\t')
    #print(t, end='\t')
    try:
        min_r = float(t[0][0])
        max_r = t[0][1]
        if max_r.startswith('(') or max_r.startswith('（'):
            max_r = float(max_r[1:-1])
        else:
            max_r = float(max_r)
    except:
        min_r = -(float(t[0][0].split('-')[1]))
        max_r = -(float(t[0][1][:-1]))
        #print('**', end='\t')
    for j in range(length):
        if float(df.iloc[j, i]) <= min_r or float(df.iloc[j, i]) >= max_r:
            df.iloc[j, i] = np.nan

df.to_excel('样本数据 a.xlsx')

#2-2
#2/8 法则删除空值超过 20%的数据
```

```
failist = []
dfb = pd.read_excel('样本数据 b.xlsx')
columns = list(dfb.columns)
length = dfb.shape[0]
fail_limit = int(length * 0.2)
for i in range(len(columns)):
    n = 0
    for j in range(length):
        if str(dfb.iloc[j, i]) == 'nan':
            n += 1
    if n > fail_limit:
        failist.append(columns[i])
dfb = dfb.drop(failist, axis=1)

dfb.to_excel('样本数据 c.xlsx')

#2-3
#根据 3sigma 法则处理非操作变量
df_fc = pd.read_excel('非操作变量数据.xlsx')
max_list = df_fc.mean().values + 3 * df_fc.std()
min_list = df_fc.mean().values - 3 * df_fc.std()
columns = list(df_fc.columns)
print(columns)
for i in range(len(columns)):
    for j in range(length):
        if float(df_fc.iloc[j, i]) > max_list[i] or float(df_fc.iloc[j, i]) < min_list[i]:
            df_fc.iloc[j, i] = np.nan
df_fc.to_excel('非操作变量数据 a.xlsx')

#2-5
#反距离权重法处理空缺值
df = pd.read_excel('样本数据 c.xlsx')
columns = list(df.columns)
length = df.shape[0]
c = defaultdict(int)
df_norm_ = (df - df.min()) / (df.max() - df.min())
df_norm = df_norm_.dropna(how='any')
df_drop = df.dropna(how='any')
def anti_distance_x(df_row, df_norm, df_drop, i):
    x = 0.0
    adistance = 0.0
    #print(df_row)
    nan_index = np.where(np.isnan(df_row.values))
    print(nan_index[0])
```

```
    print(np.array(list(df_row)))
    row_array = np.delete(np.array(list(df_row)), nan_index[0])
    df_norm_without_i = df_norm.drop(df_norm.columns[nan_index[0]], axis=1)
    length = df_norm_without_i.shape[0]
    for j in range(length):
        #print(row_array)
        y_array = df_norm_without_i.iloc[j, :].values
        #print(y_array)
        distance = np.sqrt(np.sum(np.square(row_array - y_array)))
        print('=====}/{}====='.format(j, i))
        #print(row_array)
        #print(y_array)
        #print(distance)
        c = df_drop.iloc[j, i] / distance / distance
        print(c)
        x += c
        adistance += (1 / (distance ** 2))
        print(df_drop.iloc[j, i])
    print(x/adistance)
    return x/adistance
tt = 0
for j in range(length):
    for i in range(len(columns)):
        if str(df.iloc[j, i]) == 'nan':
            tt += 1
            df.iloc[j, i] = anti_distance_x(df_norm_.iloc[j, :], df_norm, df_drop, i)

print(tt)
df_drop.to_excel('drop.xlsx')
df.to_excel('样本数据 z_反距离权重法.xlsx')
```

问题二的 SPSS 代码

```
#2-1
covariance
  /VARIABLES=SZORB.CAL_H2.PV SZORB.PDI_2102.PV SZORB.PT_2801.PV
SZORB.FC_2801.PV SZORB.TE_2103.PV
    SZORB.TE_2005.PV SZORB.PT_2101.PV SZORB.PDT_2104.PV
SZORB.SIS_PDT_2103B.PV SZORB.TC_2101.PV
    SZORB.TE_2301.PV SZORB.PT_2301.PV SZORB.PC_2105.PV SZORB.PC_5101.PV
SZORB.TC_5005.PV
    SZORB.LC_5001.PV SZORB.LC_5101.PV SZORB.TE_5102.PV SZORB.TE_5202.PV
SZORB.FC_5202.PV
    SZORB.PT_9301.PV SZORB.FT_9301.PV SZORB.FT_5101.PV SZORB.TE_9001.PV
SZORB.PT_9001.PV
```

SZORB.FT_9001.PV SZORB.FT_9403.PV SZORB.PT_9403.PV SZORB.TE_9301.PV SZORB.FT_9201.PV
SZORB.FT_9202.PV SZORB.FT_9302.PV SZORB.FT_3301.PV SZORB.FT_9402.PV SZORB.PT_9402.PV
SZORB.FT_9401.PV SZORB.PT_9401.PV SZORB.PDC_2502.PV SZORB.FC_2501.PV SZORB.FT_1001.PV
SZORB.FT_1003.PV SZORB.FT_1004.PV SZORB.TE_1001.PV SZORB.FC_1005.PV SZORB.FC_1101.PV
SZORB.FC_1102.PV SZORB.AT_1001.PV SZORB.TE_1105.PV SZORB.PDI_1102.PV SZORB.TE_1601.PV
SZORB.SIS_TE_6010.PV SZORB.PC_6001.PV SZORB.AC_6001.PV SZORB.TE_1608.PV SZORB.TC_1606.PV
SZORB.PT_6002.PV SZORB.PC_1603.PV SZORB.PT_1602A.PV SZORB.PC_1301.PV SZORB.PT_1201.PV
SZORB.LC_1201.PV SZORB.FC_1201.PV SZORB.TE_1201.PV SZORB.TE_1203.PV SZORB.LC_1202.PV
SZORB.FC_1203.PV SZORB.PC_1202.PV SZORB.TC_2801.PV SZORB.FC_3101.PV SZORB.FC_2601.PV
SZORB.PC_2601.PV SZORB.PDT_2604.PV SZORB.TE_2601.PV SZORB.TC_2607.PV SZORB.PDI_2703A.PV
SZORB.PDC_2607.PV SZORB.FT_9102.PV SZORB.PT_1501.PV SZORB.FT_1004.TOTAL SZORB.FT_9001.TOTAL
SZORB.FT_5104.TOTAL SZORB.FT_5201.TOTAL SZORB.FT_5101.TOTAL SZORB.FT_9101.TOTAL SZORB.FT_1003.TOTAL
SZORB.FT_3301.TOTAL SZORB.FT_9201.TOTAL SZORB.FT_9202.TOTAL SZORB.FT_9301.TOTAL SZORB.FT_9302.TOTAL
SZORB.FT_9401.TOTAL SZORB.FT_9402.TOTAL SZORB.FT_9403.TOTAL SZORB.FT_1202.TOTAL SZORB.FT_5201.PV
SZORB.FC_1101.TOTAL SZORB.FT_1204.PV SZORB.FT_5102.TOTAL SZORB.FC_1202.TOTAL SZORB.FT_9102.TOTAL
SZORB.FT_1001.TOTAL SZORB.TE_1101.DACA SZORB.PT_1102.DACA SZORB.PT_1103.DACA SZORB.TE_1104.DACA
SZORB.TE_1107.DACA SZORB.TE_1103.DACA SZORB.TE_1106.DACA SZORB.LI_9102.DACA SZORB.TE_9003.DACA
SZORB.TE_9002.DACA SZORB.FT_9002.DACA SZORB.PC_9002.DACA SZORB.LT_9001.DACA SZORB.LC_5002.DACA
SZORB.LC_5102.DACA SZORB.LT_3801.DACA SZORB.LT_3101.DACA SZORB.PC_3101.DACA SZORB.TE_3101.DACA
SZORB.FT_3303.DACA SZORB.LC_3301.DACA SZORB.PC_3301.DACA SZORB.FT_3304.DACA SZORB.LT_1501.DACA
SZORB.TE_1501.DACA SZORB.TE_1502.DACA SZORB.LC_1203.DACA SZORB.LT_2101.DACA SZORB.FT_3001.DACA
SZORB.FT_2701.DACA SZORB.SIS_PT_2703 SZORB.FC_2702.DACA SZORB.TC_2702.DACA SZORB.PT_2905.DACA

SZORB.LT_2901.DACA SZORB.TE_2901.DACA SZORB.TE_2902.DACA SZORB.FT_2502.DACA SZORB.TE_2501.DACA
SZORB.PT_2501.DACA SZORB.PT_2502.DACA SZORB.PDT_2503.DACA SZORB.ZT_2533.DACA SZORB.FT_2433.DACA
SZORB.TE_2401.DACA SZORB.FC_2432.DACA SZORB.FT_2303.DACA SZORB.FT_2302.DACA SZORB.LT_1301.DACA
SZORB.SIS_TE_2802 SZORB.LT_1002.DACA SZORB.TE_5002.DACA SZORB.TE_5004.DACA SZORB.FC_5203.DACA
SZORB.TE_5006.DACA SZORB.TE_5003.DACA SZORB.TE_5201.DACA SZORB.TE_5101.DACA SZORB.FT_2431.DACA
SZORB.TC_2201.PV SZORB.TC_2201.OP SZORB.FT_3201.DACA SZORB.SIS_PT_2602.PV SZORB.SIS_TE_2606.PV
SZORB.SIS_TE_2605.PV SZORB.PDT_2704.DACA SZORB.PDT_2703B.DACA SZORB.PDC_2702.DACA
SZORB.PDI_2501.DACA SZORB.AT_1001.DACA SZORB.PT_6009.DACA SZORB.LI_2107.DACA SZORB.LI_2104.DACA
SZORB.TE_6002.DACA SZORB.TE_6001.DACA SZORB.PT_1101.DACA SZORB.FT_3501.DACA SZORB.PC_3001.DACA
SZORB.FC_5103.DACA SZORB.TE_5001.DACA SZORB.FT_2002.DACA SZORB.PDT_3601.DACA SZORB.PDT_3602.DACA
SZORB.PT_6006.DACA SZORB.SIS_TE_6009.PV SZORB.SIS_PT_6007.PV SZORB.TE_6008.DACA SZORB.PT_5201.DACA
SZORB.PC_3501.DACA SZORB.FT_2001.DACA SZORB.LT_9101.DACA SZORB.PDI_2801.DACA SZORB.PT_6003.DACA
SZORB.AT_6201.DACA SZORB.PDI_2301.DACA SZORB.PDI_2105.DACA SZORB.PC_2902.DACA SZORB.BS_LT_2401.PV
SZORB.BS_AT_2402.PV SZORB.PC_2401.DACA SZORB.BS_AT_2401.PV SZORB.PC_2401B.DACA SZORB.FT_3701.DACA
SZORB.FT_3702.DACA SZORB.PT_2603.DACA SZORB.LC_2601.DACA SZORB.PDT_2605.DACA SZORB.PT_2607.DACA
SZORB.PDT_2606.DACA SZORB.ZT_2634.DACA SZORB.TE_2608.DACA SZORB.TE_2603.DACA SZORB.TE_2604.DACA
SZORB.DT_2001.DACA SZORB.DT_2107.DACA SZORB.TE_2104.DACA SZORB.PDT_2001.DACA SZORB.TE_2002.DACA
SZORB.TE_2001.DACA SZORB.TE_2004.DACA SZORB.TE_2003.DACA SZORB.PC_2401B.PIDA.SP
SZORB.PC_2401B.PIDA.OP SZORB.PC_2401.PIDA.OP SZORB.PC_2401.PIDA.SP SZORB.FT_3302.DACA
SZORB.PDT_1003.DACA SZORB.PDT_1002.DACA SZORB.PDT_2409.DACA SZORB.PDT_3503.DACA SZORB.PDT_3502.DACA
SZORB.PDT_2906.DACA SZORB.PDT_3002.DACA SZORB.PDT_1004.DACA SZORB.PDI_2903.DACA SZORB.PT_2901.DACA
SZORB.PT_2106.DACA SZORB.FT_1301.DACA SZORB.PT_7510B.DACA SZORB.TE_7508B.DACA SZORB.PT_7508B.DACA

```
    SZORB.TE_7506B.DACA SZORB.PT_7510.DACA SZORB.TE_7508.DACA
SZORB.PT_7508.DACA SZORB.TE_7506.DACA
    SZORB.PT_7505B.DACA SZORB.TE_7504B.DACA SZORB.PT_7503B.DACA
SZORB.TE_7502B.DACA SZORB.PT_7505.DACA
    SZORB.TE_7504.DACA SZORB.PT_7503.DACA SZORB.PT_7502.DACA
SZORB.TE_7106B.DACA SZORB.TE_7108B.DACA
    SZORB.PT_7107B.DACA SZORB.PT_7103B.DACA SZORB.TE_7102B.DACA
SZORB.TE_7106.DACA SZORB.PT_7107.DACA
    SZORB.PT_7103.DACA SZORB.TE_7102.DACA
SZORB.HIC_2533.AUTOMANA.OP SZORB.FC_2432.PIDA.SP
    SZORB.PT_1604.DACA SZORB.TC_1607.DACA SZORB.PT_6005.DACA
SZORB.PT_6008.DACA SZORB.PT_1601.DACA
    SZORB.TE_1605.DACA SZORB.TE_1604.DACA SZORB.TE_1603.DACA
SZORB.TE_1602.DACA SZORB.SIS_FT_3202.PV
    SZORB.TXE_3202A.DACA SZORB.TXE_3201A.DACA SZORB.TC_3203.DACA
SZORB.SIS_TEX_3103B.PV
    SZORB.TE_3111.DACA SZORB.TE_3112.DACA SZORB.TXE_2203A.DACA
SZORB.TXE_2202A.DACA SZORB.TE_5008.DACA
    SZORB.TE_5009.DACA SZORB.FC_5001.DACA SZORB.TE_5007.DACA
SZORB.TE_1504.DACA SZORB.TE_1503.DACA
    SZORB.TC_3102.DACA SZORB.TE_1102.DACA SZORB.AT0001.DACA.PV
SZORB.AT0002.DACA.PV
    SZORB.AT0003.DACA.PV SZORB.AT0004.DACA.PV SZORB.AT0005.DACA.PV
SZORB.AT0006.DACA.PV
    SZORB.AT0007.DACA.PV SZORB.AT0008.DACA.PV SZORB.AT0009.DACA.PV
SZORB.AT0010.DACA.PV
    SZORB.AT0011.DACA.PV SZORB.AT0012.DACA.PV SZORB.AT0013.DACA.PV
SZORB.TE_2104.DACA.PV
    SZORB.SIS_PDT_2103A.PV SZORB.PT_2106.DACA.PV
SZORB.TE_6008.DACA.PV SZORB.TE_6001.DACA.PV
    SZORB.FT_1204.DACA.PV SZORB.LC_1203.PIDA.PV SZORB.LC_5102.PIDA.PV
SZORB.TE_1103.DACA.PV
    SZORB.TE_1104.DACA.PV SZORB.TE_1102.DACA.PV SZORB.TE_1106.DACA.PV
SZORB.TE_1107.DACA.PV
    SZORB.TE_1101.DACA.PV SZORB.CAL.LINE.PV SZORB.CAL.CANGLIANG.PV
SZORB.CAL.SPEED.PV
    SZORB.CAL.LEVEL.PV SZORB.RXL_0001.AUXCALCA.PV
SZORB.CAL_1.CANGLIANG.PV SZORB.FT_1006.DACA.PV
    SZORB.FT_1006.TOTALIZERA.PV SZORB.FT_5204.TOTALIZERA.PV
SZORB.FT_1503.DACA.PV
    SZORB.FT_1503.TOTALIZERA.PV SZORB.FT_1504.DACA.PV
SZORB.FT_1504.TOTALIZERA.PV SZORB.PC_1001A.PV
  /PRINT=SPEARMAN TWOTAIL SIG
  /MISSING=PAIRWISE.
```

```
REGRESSION
  /MISSING LISTWISE
  /STATISTICS COEFF OUTS CI(99) R ANOVA CHANGE
  /CRITERIA=PIN(.05) POUT(.10)
  /NOORIGIN
  /DEPENDENT value
  /METHOD=STEPWISE 辛烷值 RON 饱和烃 v（烷烃环烷烃） 烯烃 v 芳烃 v 溴值
gBr100g 密度 20℃kgm³ 硫含量 μgg.1 焦炭 wt Swt 焦炭 wt.1 Swt.1
    SZORB.CAL_H2.PV SZORB.PDI_2102.PV SZORB.PT_2801.PV SZORB.FC_2801.PV
SZORB.TE_2103.PV
    SZORB.TE_2005.PV SZORB.PT_2101.PV SZORB.PDT_2104.PV
SZORB.SIS_PDT_2103B.PV SZORB.TC_2101.PV
    SZORB.TE_2301.PV SZORB.PT_2301.PV SZORB.PC_2105.PV SZORB.PC_5101.PV
SZORB.TC_5005.PV
    SZORB.LC_5001.PV SZORB.LC_5101.PV SZORB.TE_5102.PV SZORB.TE_5202.PV
SZORB.FC_5202.PV
    SZORB.PT_9301.PV SZORB.FT_9301.PV SZORB.FT_5101.PV SZORB.TE_9001.PV
SZORB.PT_9001.PV
    SZORB.FT_9001.PV SZORB.FT_9403.PV SZORB.PT_9403.PV SZORB.TE_9301.PV
SZORB.FT_9201.PV
    SZORB.FT_9202.PV SZORB.FT_9302.PV SZORB.FT_3301.PV SZORB.FT_9402.PV
SZORB.PT_9402.PV
    SZORB.FT_9401.PV SZORB.PT_9401.PV SZORB.PDC_2502.PV
SZORB.FC_2501.PV SZORB.FT_1001.PV
    SZORB.FT_1003.PV SZORB.FT_1004.PV SZORB.TE_1001.PV SZORB.FC_1005.PV
SZORB.FC_1101.PV
    SZORB.FC_1102.PV SZORB.AT_1001.PV SZORB.TE_1105.PV
SZORB.PDI_1102.PV SZORB.TE_1601.PV
    SZORB.SIS_TE_6010.PV SZORB.PC_6001.PV SZORB.AC_6001.PV
SZORB.TE_1608.PV SZORB.TC_1606.PV
    SZORB.PT_6002.PV SZORB.PC_1603.PV SZORB.PT_1602A.PV
SZORB.PC_1301.PV SZORB.PT_1201.PV
    SZORB.LC_1201.PV SZORB.FC_1201.PV SZORB.TE_1201.PV SZORB.TE_1203.PV
SZORB.LC_1202.PV
    SZORB.FC_1203.PV SZORB.PC_1202.PV SZORB.TC_2801.PV SZORB.FC_3101.PV
SZORB.FC_2601.PV
    SZORB.PC_2601.PV SZORB.PDT_2604.PV SZORB.TE_2601.PV
SZORB.TC_2607.PV SZORB.PDI_2703A.PV
    SZORB.PDC_2607.PV SZORB.FT_9102.PV SZORB.PT_1501.PV
SZORB.FT_1004.TOTAL SZORB.FT_9001.TOTAL
    SZORB.FT_5104.TOTAL SZORB.FT_5201.TOTAL SZORB.FT_5101.TOTAL
SZORB.FT_9101.TOTAL SZORB.FT_1003.TOTAL
    SZORB.FT_3301.TOTAL SZORB.FT_9201.TOTAL SZORB.FT_9202.TOTAL
SZORB.FT_9301.TOTAL SZORB.FT_9302.TOTAL
```

SZORB.FT_9401.TOTAL SZORB.FT_9402.TOTAL SZORB.FT_9403.TOTAL SZORB.FT_1202.TOTAL SZORB.FT_5201.PV
SZORB.FC_1101.TOTAL SZORB.FT_1204.PV SZORB.FT_5102.TOTAL SZORB.FC_1202.TOTAL SZORB.FT_9102.TOTAL
SZORB.FT_1001.TOTAL SZORB.TE_1101.DACA SZORB.PT_1102.DACA SZORB.PT_1103.DACA SZORB.TE_1104.DACA
SZORB.TE_1107.DACA SZORB.TE_1103.DACA SZORB.TE_1106.DACA SZORB.LI_9102.DACA SZORB.TE_9003.DACA
SZORB.TE_9002.DACA SZORB.FT_9002.DACA SZORB.PC_9002.DACA SZORB.LT_9001.DACA SZORB.LC_5002.DACA
SZORB.LC_5102.DACA SZORB.LT_3801.DACA SZORB.LT_3101.DACA SZORB.PC_3101.DACA SZORB.TE_3101.DACA
SZORB.FT_3303.DACA SZORB.LC_3301.DACA SZORB.PC_3301.DACA SZORB.FT_3304.DACA SZORB.LT_1501.DACA
SZORB.TE_1501.DACA SZORB.TE_1502.DACA SZORB.LC_1203.DACA SZORB.LT_2101.DACA SZORB.FT_3001.DACA
SZORB.FT_2701.DACA SZORB.SIS_PT_2703 SZORB.FC_2702.DACA SZORB.TC_2702.DACA SZORB.PT_2905.DACA
SZORB.LT_2901.DACA SZORB.TE_2901.DACA SZORB.TE_2902.DACA SZORB.FT_2502.DACA SZORB.TE_2501.DACA
SZORB.PT_2501.DACA SZORB.PT_2502.DACA SZORB.PDT_2503.DACA SZORB.ZT_2533.DACA SZORB.FT_2433.DACA
SZORB.TE_2401.DACA SZORB.FC_2432.DACA SZORB.FT_2303.DACA SZORB.FT_2302.DACA SZORB.LT_1301.DACA
SZORB.SIS_TE_2802 SZORB.LT_1002.DACA SZORB.TE_5002.DACA SZORB.TE_5004.DACA SZORB.FC_5203.DACA
SZORB.TE_5006.DACA SZORB.TE_5003.DACA SZORB.TE_5201.DACA SZORB.TE_5101.DACA SZORB.FT_2431.DACA
SZORB.TC_2201.PV SZORB.TC_2201.OP SZORB.FT_3201.DACA SZORB.SIS_PT_2602.PV SZORB.SIS_TE_2606.PV
SZORB.SIS_TE_2605.PV SZORB.PDT_2704.DACA SZORB.PDT_2703B.DACA SZORB.PDC_2702.DACA
SZORB.PDI_2501.DACA SZORB.AT_1001.DACA SZORB.PT_6009.DACA SZORB.LI_2107.DACA SZORB.LI_2104.DACA
SZORB.TE_6002.DACA SZORB.TE_6001.DACA SZORB.PT_1101.DACA SZORB.FT_3501.DACA SZORB.PC_3001.DACA
SZORB.FC_5103.DACA SZORB.TE_5001.DACA SZORB.FT_2002.DACA SZORB.PDT_3601.DACA SZORB.PDT_3602.DACA
SZORB.PT_6006.DACA SZORB.SIS_TE_6009.PV SZORB.SIS_PT_6007.PV SZORB.TE_6008.DACA SZORB.PT_5201.DACA
SZORB.PC_3501.DACA SZORB.FT_2001.DACA SZORB.LT_9101.DACA SZORB.PDI_2801.DACA SZORB.PT_6003.DACA
SZORB.AT_6201.DACA SZORB.PDI_2301.DACA SZORB.PDI_2105.DACA SZORB.PC_2902.DACA SZORB.BS_LT_2401.PV

SZORB.BS_AT_2402.PV SZORB.PC_2401.DACA SZORB.BS_AT_2401.PV SZORB.PC_2401B.DACA SZORB.FT_3701.DACA
SZORB.FT_3702.DACA SZORB.PT_2603.DACA SZORB.LC_2601.DACA SZORB.PDT_2605.DACA SZORB.PT_2607.DACA
SZORB.PDT_2606.DACA SZORB.ZT_2634.DACA SZORB.TE_2608.DACA SZORB.TE_2603.DACA SZORB.TE_2604.DACA
SZORB.DT_2001.DACA SZORB.DT_2107.DACA SZORB.TE_2104.DACA SZORB.PDT_2001.DACA SZORB.TE_2002.DACA
SZORB.TE_2001.DACA SZORB.TE_2004.DACA SZORB.TE_2003.DACA SZORB.PC_2401B.PIDA.SP
SZORB.PC_2401B.PIDA.OP SZORB.PC_2401.PIDA.OP SZORB.PC_2401.PIDA.SP SZORB.FT_3302.DACA
SZORB.PDT_1003.DACA SZORB.PDT_1002.DACA SZORB.PDT_2409.DACA SZORB.PDT_3503.DACA SZORB.PDT_3502.DACA
SZORB.PDT_2906.DACA SZORB.PDT_3002.DACA SZORB.PDT_1004.DACA SZORB.PDI_2903.DACA SZORB.PT_2901.DACA
SZORB.PT_2106.DACA SZORB.FT_1301.DACA SZORB.PT_7510B.DACA SZORB.TE_7508B.DACA SZORB.PT_7508B.DACA
SZORB.TE_7506B.DACA SZORB.PT_7510.DACA SZORB.TE_7508.DACA SZORB.PT_7508.DACA SZORB.TE_7506.DACA
SZORB.PT_7505B.DACA SZORB.TE_7504B.DACA SZORB.PT_7503B.DACA SZORB.TE_7502B.DACA SZORB.PT_7505.DACA
SZORB.TE_7504.DACA SZORB.PT_7503.DACA SZORB.PT_7502.DACA SZORB.TE_7106B.DACA SZORB.TE_7108B.DACA
SZORB.PT_7107B.DACA SZORB.PT_7103B.DACA SZORB.TE_7102B.DACA SZORB.TE_7106.DACA SZORB.PT_7107.DACA
SZORB.PT_7103.DACA SZORB.TE_7102.DACA SZORB.HIC_2533.AUTOMANA.OP SZORB.FC_2432.PIDA.SP
SZORB.PT_1604.DACA SZORB.TC_1607.DACA SZORB.PT_6005.DACA SZORB.PT_6008.DACA SZORB.PT_1601.DACA
SZORB.TE_1605.DACA SZORB.TE_1604.DACA SZORB.TE_1603.DACA SZORB.TE_1602.DACA SZORB.SIS_FT_3202.PV
SZORB.TXE_3202A.DACA SZORB.TXE_3201A.DACA SZORB.TC_3203.DACA SZORB.SIS_TEX_3103B.PV
SZORB.TE_3111.DACA SZORB.TE_3112.DACA SZORB.TXE_2203A.DACA SZORB.TXE_2202A.DACA SZORB.TE_5008.DACA
SZORB.TE_5009.DACA SZORB.FC_5001.DACA SZORB.TE_5007.DACA SZORB.TE_1504.DACA SZORB.TE_1503.DACA
SZORB.TC_3102.DACA SZORB.TE_1102.DACA SZORB.AT0001.DACA.PV SZORB.AT0002.DACA.PV
SZORB.AT0003.DACA.PV SZORB.AT0004.DACA.PV SZORB.AT0005.DACA.PV SZORB.AT0006.DACA.PV
SZORB.AT0007.DACA.PV SZORB.AT0008.DACA.PV SZORB.AT0009.DACA.PV SZORB.AT0010.DACA.PV

SZORB.AT0011.DACA.PV SZORB.AT0012.DACA.PV SZORB.AT0013.DACA.PV SZORB.TE_2104.DACA.PV
SZORB.SIS_PDT_2103A.PV SZORB.PT_2106.DACA.PV SZORB.TE_6008.DACA.PV SZORB.TE_6001.DACA.PV
SZORB.FT_1204.DACA.PV SZORB.LC_1203.PIDA.PV SZORB.LC_5102.PIDA.PV SZORB.TE_1103.DACA.PV
SZORB.TE_1104.DACA.PV SZORB.TE_1102.DACA.PV SZORB.TE_1106.DACA.PV SZORB.TE_1107.DACA.PV
SZORB.TE_1101.DACA.PV SZORB.CAL.LINE.PV SZORB.CAL.CANGLIANG.PV SZORB.CAL.SPEED.PV
SZORB.CAL.LEVEL.PV SZORB.RXL_0001.AUXCALCA.PV SZORB.CAL_1.CANGLIANG.PV SZORB.FT_1006.DACA.PV
SZORB.FT_1006.TOTALIZERA.PV SZORB.FT_5204.TOTALIZERA.PV SZORB.FT_1503.DACA.PV
SZORB.FT_1503.TOTALIZERA.PV SZORB.FT_1504.DACA.PV SZORB.FT_1504.TOTALIZERA.PV SZORB.PC_1001A.PV
/RESIDUALS DURBIN.

#2-2
FACTOR
/VARIABLES 辛烷值 RON 饱和烃 v（烷烃环烷烃） 烯烃 v 芳烃 v 溴值 gBr100g 密度 20℃kgm³ 硫含量 μgg.1 焦炭 wt Swt 焦炭 wt.1 Swt.1
SZORB.CAL_H2.PV SZORB.PDI_2102.PV SZORB.PT_2801.PV SZORB.FC_2801.PV SZORB.TE_2103.PV
SZORB.TE_2005.PV SZORB.PT_2101.PV SZORB.PDT_2104.PV SZORB.SIS_PDT_2103B.PV SZORB.TC_2101.PV
SZORB.TE_2301.PV SZORB.PT_2301.PV SZORB.PC_2105.PV SZORB.PC_5101.PV SZORB.TC_5005.PV
SZORB.LC_5001.PV SZORB.LC_5101.PV SZORB.TE_5102.PV SZORB.TE_5202.PV SZORB.FC_5202.PV
SZORB.PT_9301.PV SZORB.FT_9301.PV SZORB.FT_5101.PV SZORB.TE_9001.PV SZORB.PT_9001.PV
SZORB.FT_9001.PV SZORB.FT_9403.PV SZORB.PT_9403.PV SZORB.TE_9301.PV SZORB.FT_9201.PV
SZORB.FT_9202.PV SZORB.FT_9302.PV SZORB.FT_3301.PV SZORB.FT_9402.PV SZORB.PT_9402.PV
SZORB.FT_9401.PV SZORB.PT_9401.PV SZORB.PDC_2502.PV SZORB.FC_2501.PV SZORB.FT_1001.PV
SZORB.FT_1003.PV SZORB.FT_1004.PV SZORB.TE_1001.PV SZORB.FC_1005.PV SZORB.FC_1101.PV
SZORB.FC_1102.PV SZORB.AT_1001.PV SZORB.TE_1105.PV SZORB.PDI_1102.PV SZORB.TE_1601.PV
SZORB.SIS_TE_6010.PV SZORB.PC_6001.PV SZORB.AC_6001.PV SZORB.TE_1608.PV SZORB.TC_1606.PV

SZORB.PT_6002.PV SZORB.PC_1603.PV SZORB.PT_1602A.PV SZORB.PC_1301.PV SZORB.PT_1201.PV
SZORB.LC_1201.PV SZORB.FC_1201.PV SZORB.TE_1201.PV SZORB.TE_1203.PV SZORB.LC_1202.PV
SZORB.FC_1203.PV SZORB.PC_1202.PV SZORB.TC_2801.PV SZORB.FC_3101.PV SZORB.FC_2601.PV
SZORB.PC_2601.PV SZORB.PDT_2604.PV SZORB.TE_2601.PV SZORB.TC_2607.PV SZORB.PDI_2703A.PV
SZORB.PDC_2607.PV SZORB.FT_9102.PV SZORB.PT_1501.PV SZORB.FT_1004.TOTAL SZORB.FT_9001.TOTAL
SZORB.FT_5104.TOTAL SZORB.FT_5201.TOTAL SZORB.FT_5101.TOTAL SZORB.FT_9101.TOTAL SZORB.FT_1003.TOTAL
SZORB.FT_3301.TOTAL SZORB.FT_9201.TOTAL SZORB.FT_9202.TOTAL SZORB.FT_9301.TOTAL SZORB.FT_9302.TOTAL
SZORB.FT_9401.TOTAL SZORB.FT_9402.TOTAL SZORB.FT_9403.TOTAL SZORB.FT_1202.TOTAL SZORB.FT_5201.PV
SZORB.FC_1101.TOTAL SZORB.FT_1204.PV SZORB.FT_5102.TOTAL SZORB.FC_1202.TOTAL SZORB.FT_9102.TOTAL
SZORB.FT_1001.TOTAL SZORB.TE_1101.DACA SZORB.PT_1102.DACA SZORB.PT_1103.DACA SZORB.TE_1104.DACA
SZORB.TE_1107.DACA SZORB.TE_1103.DACA SZORB.TE_1106.DACA SZORB.LI_9102.DACA SZORB.TE_9003.DACA
SZORB.TE_9002.DACA SZORB.FT_9002.DACA SZORB.PC_9002.DACA SZORB.LT_9001.DACA SZORB.LC_5002.DACA
SZORB.LC_5102.DACA SZORB.LT_3801.DACA SZORB.LT_3101.DACA SZORB.PC_3101.DACA SZORB.TE_3101.DACA
SZORB.FT_3303.DACA SZORB.LC_3301.DACA SZORB.PC_3301.DACA SZORB.FT_3304.DACA SZORB.LT_1501.DACA
SZORB.TE_1501.DACA SZORB.TE_1502.DACA SZORB.LC_1203.DACA SZORB.LT_2101.DACA SZORB.FT_3001.DACA
SZORB.FT_2701.DACA SZORB.SIS_PT_2703 SZORB.FC_2702.DACA SZORB.TC_2702.DACA SZORB.PT_2905.DACA
SZORB.LT_2901.DACA SZORB.TE_2901.DACA SZORB.TE_2902.DACA SZORB.FT_2502.DACA SZORB.TE_2501.DACA
SZORB.PT_2501.DACA SZORB.PT_2502.DACA SZORB.PDT_2503.DACA SZORB.ZT_2533.DACA SZORB.FT_2433.DACA
SZORB.TE_2401.DACA SZORB.FC_2432.DACA SZORB.FT_2303.DACA SZORB.FT_2302.DACA SZORB.LT_1301.DACA
SZORB.SIS_TE_2802 SZORB.LT_1002.DACA SZORB.TE_5002.DACA SZORB.TE_5004.DACA SZORB.FC_5203.DACA
SZORB.TE_5006.DACA SZORB.TE_5003.DACA SZORB.TE_5201.DACA SZORB.TE_5101.DACA SZORB.FT_2431.DACA
SZORB.TC_2201.PV SZORB.TC_2201.OP SZORB.FT_3201.DACA SZORB.SIS_PT_2602.PV SZORB.SIS_TE_2606.PV

SZORB.SIS_TE_2605.PV SZORB.PDT_2704.DACA SZORB.PDT_2703B.DACA SZORB.PDC_2702.DACA
SZORB.PDI_2501.DACA SZORB.AT_1001.DACA SZORB.PT_6009.DACA SZORB.LI_2107.DACA SZORB.LI_2104.DACA
SZORB.TE_6002.DACA SZORB.TE_6001.DACA SZORB.PT_1101.DACA SZORB.FT_3501.DACA SZORB.PC_3001.DACA
SZORB.FC_5103.DACA SZORB.TE_5001.DACA SZORB.FT_2002.DACA SZORB.PDT_3601.DACA SZORB.PDT_3602.DACA
SZORB.PT_6006.DACA SZORB.SIS_TE_6009.PV SZORB.SIS_PT_6007.PV SZORB.TE_6008.DACA SZORB.PT_5201.DACA
SZORB.PC_3501.DACA SZORB.FT_2001.DACA SZORB.LT_9101.DACA SZORB.PDI_2801.DACA SZORB.PT_6003.DACA
SZORB.AT_6201.DACA SZORB.PDI_2301.DACA SZORB.PDI_2105.DACA SZORB.PC_2902.DACA SZORB.BS_LT_2401.PV
SZORB.BS_AT_2402.PV SZORB.PC_2401.DACA SZORB.BS_AT_2401.PV SZORB.PC_2401B.DACA SZORB.FT_3701.DACA
SZORB.FT_3702.DACA SZORB.PT_2603.DACA SZORB.LC_2601.DACA SZORB.PDT_2605.DACA SZORB.PT_2607.DACA
SZORB.PDT_2606.DACA SZORB.ZT_2634.DACA SZORB.TE_2608.DACA SZORB.TE_2603.DACA SZORB.TE_2604.DACA
SZORB.DT_2001.DACA SZORB.DT_2107.DACA SZORB.TE_2104.DACA SZORB.PDT_2001.DACA SZORB.TE_2002.DACA
SZORB.TE_2001.DACA SZORB.TE_2004.DACA SZORB.TE_2003.DACA SZORB.PC_2401B.PIDA.SP
SZORB.PC_2401B.PIDA.OP SZORB.PC_2401.PIDA.OP SZORB.PC_2401.PIDA.SP SZORB.FT_3302.DACA
SZORB.PDT_1003.DACA SZORB.PDT_1002.DACA SZORB.PDT_2409.DACA SZORB.PDT_3503.DACA SZORB.PDT_3502.DACA
SZORB.PDT_2906.DACA SZORB.PDT_3002.DACA SZORB.PDT_1004.DACA SZORB.PDI_2903.DACA SZORB.PT_2901.DACA
SZORB.PT_2106.DACA SZORB.FT_1301.DACA SZORB.PT_7510B.DACA SZORB.TE_7508B.DACA SZORB.PT_7508B.DACA
SZORB.TE_7506B.DACA SZORB.PT_7510.DACA SZORB.TE_7508.DACA SZORB.PT_7508.DACA SZORB.TE_7506.DACA
SZORB.PT_7505B.DACA SZORB.TE_7504B.DACA SZORB.PT_7503B.DACA SZORB.TE_7502B.DACA SZORB.PT_7505.DACA
SZORB.TE_7504.DACA SZORB.PT_7503.DACA SZORB.PT_7502.DACA SZORB.TE_7106B.DACA SZORB.TE_7108B.DACA
SZORB.PT_7107B.DACA SZORB.PT_7103B.DACA SZORB.TE_7102B.DACA SZORB.TE_7106.DACA SZORB.PT_7107.DACA
SZORB.PT_7103.DACA SZORB.TE_7102.DACA SZORB.HIC_2533.AUTOMANA.OP SZORB.FC_2432.PIDA.SP
SZORB.PT_1604.DACA SZORB.TC_1607.DACA SZORB.PT_6005.DACA SZORB.PT_6008.DACA SZORB.PT_1601.DACA

SZORB.TE_1605.DACA SZORB.TE_1604.DACA SZORB.TE_1603.DACA SZORB.TE_1602.DACA SZORB.SIS_FT_3202.PV
SZORB.TXE_3202A.DACA SZORB.TXE_3201A.DACA SZORB.TC_3203.DACA SZORB.SIS_TEX_3103B.PV
SZORB.TE_3111.DACA SZORB.TE_3112.DACA SZORB.TXE_2203A.DACA SZORB.TXE_2202A.DACA SZORB.TE_5008.DACA
SZORB.TE_5009.DACA SZORB.FC_5001.DACA SZORB.TE_5007.DACA SZORB.TE_1504.DACA SZORB.TE_1503.DACA
SZORB.TC_3102.DACA SZORB.TE_1102.DACA SZORB.AT0001.DACA.PV SZORB.AT0002.DACA.PV
SZORB.AT0003.DACA.PV SZORB.AT0004.DACA.PV SZORB.AT0005.DACA.PV SZORB.AT0006.DACA.PV
SZORB.AT0007.DACA.PV SZORB.AT0008.DACA.PV SZORB.AT0009.DACA.PV SZORB.AT0010.DACA.PV
SZORB.AT0011.DACA.PV SZORB.AT0012.DACA.PV SZORB.AT0013.DACA.PV SZORB.TE_2104.DACA.PV
SZORB.SIS_PDT_2103A.PV SZORB.PT_2106.DACA.PV SZORB.TE_6008.DACA.PV SZORB.TE_6001.DACA.PV
SZORB.FT_1204.DACA.PV SZORB.LC_1203.PIDA.PV SZORB.LC_5102.PIDA.PV SZORB.TE_1103.DACA.PV
SZORB.TE_1104.DACA.PV SZORB.TE_1102.DACA.PV SZORB.TE_1106.DACA.PV SZORB.TE_1107.DACA.PV
SZORB.TE_1101.DACA.PV SZORB.CAL.LINE.PV SZORB.CAL.CANGLIANG.PV SZORB.CAL.SPEED.PV
SZORB.CAL.LEVEL.PV SZORB.RXL_0001.AUXCALCA.PV SZORB.CAL_1.CANGLIANG.PV SZORB.FT_1006.DACA.PV
SZORB.FT_1006.TOTALIZERA.PV SZORB.FT_5204.TOTALIZERA.PV SZORB.FT_1503.DACA.PV
SZORB.FT_1503.TOTALIZERA.PV SZORB.FT_1504.DACA.PV SZORB.FT_1504.TOTALIZERA.PV SZORB.PC_1001A.PV
/MISSING LISTWISE
/ANALYSIS 辛烷值 RON 饱和烃 v（烷烃环烷烃） 烯烃 v 芳烃 v 溴值 gBr100g 密度 20℃kgm³ 硫含量 μgg.1 焦炭 wt Swt 焦炭 wt.1 Swt.1
SZORB.CAL_H2.PV SZORB.PDI_2102.PV SZORB.PT_2801.PV SZORB.FC_2801.PV SZORB.TE_2103.PV
SZORB.TE_2005.PV SZORB.PT_2101.PV SZORB.PDT_2104.PV SZORB.SIS_PDT_2103B.PV SZORB.TC_2101.PV
SZORB.TE_2301.PV SZORB.PT_2301.PV SZORB.PC_2105.PV SZORB.PC_5101.PV SZORB.TC_5005.PV
SZORB.LC_5001.PV SZORB.LC_5101.PV SZORB.TE_5102.PV SZORB.TE_5202.PV SZORB.FC_5202.PV
SZORB.PT_9301.PV SZORB.FT_9301.PV SZORB.FT_5101.PV SZORB.TE_9001.PV SZORB.PT_9001.PV

SZORB.FT_9001.PV SZORB.FT_9403.PV SZORB.PT_9403.PV SZORB.TE_9301.PV SZORB.FT_9201.PV
SZORB.FT_9202.PV SZORB.FT_9302.PV SZORB.FT_3301.PV SZORB.FT_9402.PV SZORB.PT_9402.PV
SZORB.FT_9401.PV SZORB.PT_9401.PV SZORB.PDC_2502.PV SZORB.FC_2501.PV SZORB.FT_1001.PV
SZORB.FT_1003.PV SZORB.FT_1004.PV SZORB.TE_1001.PV SZORB.FC_1005.PV SZORB.FC_1101.PV
SZORB.FC_1102.PV SZORB.AT_1001.PV SZORB.TE_1105.PV SZORB.PDI_1102.PV SZORB.TE_1601.PV
SZORB.SIS_TE_6010.PV SZORB.PC_6001.PV SZORB.AC_6001.PV SZORB.TE_1608.PV SZORB.TC_1606.PV
SZORB.PT_6002.PV SZORB.PC_1603.PV SZORB.PT_1602A.PV SZORB.PC_1301.PV SZORB.PT_1201.PV
SZORB.LC_1201.PV SZORB.FC_1201.PV SZORB.TE_1201.PV SZORB.TE_1203.PV SZORB.LC_1202.PV
SZORB.FC_1203.PV SZORB.PC_1202.PV SZORB.TC_2801.PV SZORB.FC_3101.PV SZORB.FC_2601.PV
SZORB.PC_2601.PV SZORB.PDT_2604.PV SZORB.TE_2601.PV SZORB.TC_2607.PV SZORB.PDI_2703A.PV
SZORB.PDC_2607.PV SZORB.FT_9102.PV SZORB.PT_1501.PV SZORB.FT_1004.TOTAL SZORB.FT_9001.TOTAL
SZORB.FT_5104.TOTAL SZORB.FT_5201.TOTAL SZORB.FT_5101.TOTAL SZORB.FT_9101.TOTAL SZORB.FT_1003.TOTAL
SZORB.FT_3301.TOTAL SZORB.FT_9201.TOTAL SZORB.FT_9202.TOTAL SZORB.FT_9301.TOTAL SZORB.FT_9302.TOTAL
SZORB.FT_9401.TOTAL SZORB.FT_9402.TOTAL SZORB.FT_9403.TOTAL SZORB.FT_1202.TOTAL SZORB.FT_5201.PV
SZORB.FC_1101.TOTAL SZORB.FT_1204.PV SZORB.FT_5102.TOTAL SZORB.FC_1202.TOTAL SZORB.FT_9102.TOTAL
SZORB.FT_1001.TOTAL SZORB.TE_1101.DACA SZORB.PT_1102.DACA SZORB.PT_1103.DACA SZORB.TE_1104.DACA
SZORB.TE_1107.DACA SZORB.TE_1103.DACA SZORB.TE_1106.DACA SZORB.LI_9102.DACA SZORB.TE_9003.DACA
SZORB.TE_9002.DACA SZORB.FT_9002.DACA SZORB.PC_9002.DACA SZORB.LT_9001.DACA SZORB.LC_5002.DACA
SZORB.LC_5102.DACA SZORB.LT_3801.DACA SZORB.LT_3101.DACA SZORB.PC_3101.DACA SZORB.TE_3101.DACA
SZORB.FT_3303.DACA SZORB.LC_3301.DACA SZORB.PC_3301.DACA SZORB.FT_3304.DACA SZORB.LT_1501.DACA
SZORB.TE_1501.DACA SZORB.TE_1502.DACA SZORB.LC_1203.DACA SZORB.LT_2101.DACA SZORB.FT_3001.DACA
SZORB.FT_2701.DACA SZORB.SIS_PT_2703 SZORB.FC_2702.DACA SZORB.TC_2702.DACA SZORB.PT_2905.DACA

SZORB.LT_2901.DACA SZORB.TE_2901.DACA SZORB.TE_2902.DACA SZORB.FT_2502.DACA SZORB.TE_2501.DACA
SZORB.PT_2501.DACA SZORB.PT_2502.DACA SZORB.PDT_2503.DACA SZORB.ZT_2533.DACA SZORB.FT_2433.DACA
SZORB.TE_2401.DACA SZORB.FC_2432.DACA SZORB.FT_2303.DACA SZORB.FT_2302.DACA SZORB.LT_1301.DACA
SZORB.SIS_TE_2802 SZORB.LT_1002.DACA SZORB.TE_5002.DACA SZORB.TE_5004.DACA SZORB.FC_5203.DACA
SZORB.TE_5006.DACA SZORB.TE_5003.DACA SZORB.TE_5201.DACA SZORB.TE_5101.DACA SZORB.FT_2431.DACA
SZORB.TC_2201.PV SZORB.TC_2201.OP SZORB.FT_3201.DACA SZORB.SIS_PT_2602.PV SZORB.SIS_TE_2606.PV
SZORB.SIS_TE_2605.PV SZORB.PDT_2704.DACA SZORB.PDT_2703B.DACA SZORB.PDC_2702.DACA
SZORB.PDI_2501.DACA SZORB.AT_1001.DACA SZORB.PT_6009.DACA SZORB.LI_2107.DACA SZORB.LI_2104.DACA
SZORB.TE_6002.DACA SZORB.TE_6001.DACA SZORB.PT_1101.DACA SZORB.FT_3501.DACA SZORB.PC_3001.DACA
SZORB.FC_5103.DACA SZORB.TE_5001.DACA SZORB.FT_2002.DACA SZORB.PDT_3601.DACA SZORB.PDT_3602.DACA
SZORB.PT_6006.DACA SZORB.SIS_TE_6009.PV SZORB.SIS_PT_6007.PV SZORB.TE_6008.DACA SZORB.PT_5201.DACA
SZORB.PC_3501.DACA SZORB.FT_2001.DACA SZORB.LT_9101.DACA SZORB.PDI_2801.DACA SZORB.PT_6003.DACA
SZORB.AT_6201.DACA SZORB.PDI_2301.DACA SZORB.PDI_2105.DACA SZORB.PC_2902.DACA SZORB.BS_LT_2401.PV
SZORB.BS_AT_2402.PV SZORB.PC_2401.DACA SZORB.BS_AT_2401.PV SZORB.PC_2401B.DACA SZORB.FT_3701.DACA
SZORB.FT_3702.DACA SZORB.PT_2603.DACA SZORB.LC_2601.DACA SZORB.PDT_2605.DACA SZORB.PT_2607.DACA
SZORB.PDT_2606.DACA SZORB.ZT_2634.DACA SZORB.TE_2608.DACA SZORB.TE_2603.DACA SZORB.TE_2604.DACA
SZORB.DT_2001.DACA SZORB.DT_2107.DACA SZORB.TE_2104.DACA SZORB.PDT_2001.DACA SZORB.TE_2002.DACA
SZORB.TE_2001.DACA SZORB.TE_2004.DACA SZORB.TE_2003.DACA SZORB.PC_2401B.PIDA.SP
SZORB.PC_2401B.PIDA.OP SZORB.PC_2401.PIDA.OP SZORB.PC_2401.PIDA.SP SZORB.FT_3302.DACA
SZORB.PDT_1003.DACA SZORB.PDT_1002.DACA SZORB.PDT_2409.DACA SZORB.PDT_3503.DACA SZORB.PDT_3502.DACA
SZORB.PDT_2906.DACA SZORB.PDT_3002.DACA SZORB.PDT_1004.DACA SZORB.PDI_2903.DACA SZORB.PT_2901.DACA
SZORB.PT_2106.DACA SZORB.FT_1301.DACA SZORB.PT_7510B.DACA SZORB.TE_7508B.DACA SZORB.PT_7508B.DACA

SZORB.TE_7506B.DACA SZORB.PT_7510.DACA SZORB.TE_7508.DACA SZORB.PT_7508.DACA SZORB.TE_7506.DACA
SZORB.PT_7505B.DACA SZORB.TE_7504B.DACA SZORB.PT_7503B.DACA SZORB.TE_7502B.DACA SZORB.PT_7505.DACA
SZORB.TE_7504.DACA SZORB.PT_7503.DACA SZORB.PT_7502.DACA SZORB.TE_7106B.DACA SZORB.TE_7108B.DACA
SZORB.PT_7107B.DACA SZORB.PT_7103B.DACA SZORB.TE_7102B.DACA SZORB.TE_7106.DACA SZORB.PT_7107.DACA
SZORB.PT_7103.DACA SZORB.TE_7102.DACA SZORB.HIC_2533.AUTOMANA.OP SZORB.FC_2432.PIDA.SP
SZORB.PT_1604.DACA SZORB.TC_1607.DACA SZORB.PT_6005.DACA SZORB.PT_6008.DACA SZORB.PT_1601.DACA
SZORB.TE_1605.DACA SZORB.TE_1604.DACA SZORB.TE_1603.DACA SZORB.TE_1602.DACA SZORB.SIS_FT_3202.PV
SZORB.TXE_3202A.DACA SZORB.TXE_3201A.DACA SZORB.TC_3203.DACA SZORB.SIS_TEX_3103B.PV
SZORB.TE_3111.DACA SZORB.TE_3112.DACA SZORB.TXE_2203A.DACA SZORB.TXE_2202A.DACA SZORB.TE_5008.DACA
SZORB.TE_5009.DACA SZORB.FC_5001.DACA SZORB.TE_5007.DACA SZORB.TE_1504.DACA SZORB.TE_1503.DACA
SZORB.TC_3102.DACA SZORB.TE_1102.DACA SZORB.AT0001.DACA.PV SZORB.AT0002.DACA.PV
SZORB.AT0003.DACA.PV SZORB.AT0004.DACA.PV SZORB.AT0005.DACA.PV SZORB.AT0006.DACA.PV
SZORB.AT0007.DACA.PV SZORB.AT0008.DACA.PV SZORB.AT0009.DACA.PV SZORB.AT0010.DACA.PV
SZORB.AT0011.DACA.PV SZORB.AT0012.DACA.PV SZORB.AT0013.DACA.PV SZORB.TE_2104.DACA.PV
SZORB.SIS_PDT_2103A.PV SZORB.PT_2106.DACA.PV SZORB.TE_6008.DACA.PV SZORB.TE_6001.DACA.PV
SZORB.FT_1204.DACA.PV SZORB.LC_1203.PIDA.PV SZORB.LC_5102.PIDA.PV SZORB.TE_1103.DACA.PV
SZORB.TE_1104.DACA.PV SZORB.TE_1102.DACA.PV SZORB.TE_1106.DACA.PV SZORB.TE_1107.DACA.PV
SZORB.TE_1101.DACA.PV SZORB.CAL.LINE.PV SZORB.CAL.CANGLIANG.PV SZORB.CAL.SPEED.PV
SZORB.CAL.LEVEL.PV SZORB.RXL_0001.AUXCALCA.PV SZORB.CAL_1.CANGLIANG.PV SZORB.FT_1006.DACA.PV
SZORB.FT_1006.TOTALIZERA.PV SZORB.FT_5204.TOTALIZERA.PV SZORB.FT_1503.DACA.PV
SZORB.FT_1503.TOTALIZERA.PV SZORB.FT_1504.DACA.PV SZORB.FT_1504.TOTALIZERA.PV SZORB.PC_1001A.PV
/PRINT INITIAL KMO EXTRACTION
/CRITERIA MINEIGEN(1) ITERATE(25)

```
  /EXTRACTION PC
  /ROTATION NOROTATE
  /METHOD=CORRELATION.
```

问题二的 Python 代码

```
#第二问
#2-3
#LARSLASSOLARS
import numpy as np
import pandas as pd
from sklearn import model_selection
from sklearn.linear_model import Ridge
from sklearn.linear_model import RidgeCV
import matplotlib.pyplot as plt
import tensorflow as tf
import matplotlib.pyplot as plt  # 可视化绘制
from sklearn.linear_model import LarsLasso,LarsLassoCV,LarsLassoLarsCV,ElasticNet,
ElasticNetCV
columns = list(pd.read_excel('data_normal.xlsx').columns)
x_train = pd.read_excel('操作变量.xlsx').values[:276, :]
y_train = pd.read_excel('y.xlsx').values[:276]
x_test = pd.read_excel('操作变量.xlsx').values[276:, :]
y_test = pd.read_excel('y.xlsx').values[276:]
def LarsLassolarscv():
    global x_train, y_train, x_test, y_test
    LarsLasso = LarsLassoLarsCV().fit(x_train, y_train)
    print('---------')
    print(LarsLasso.alpha_)
    print(LarsLasso.coef_)
    return LarsLasso
LarsLassolarscv()

#2-4
#ElasticNet
def elasticnetcv():
    global x_train, y_train, x_test, y_test
    LarsLasso = ElasticNetCV().fit(x_train, y_train)
    print('---------')
    print(LarsLasso.alpha_)
    #print(LarsLasso.coef_)
    elastic = elasticnetcv()
    coef = elastic.coef_
    inte = elastic.intercept_
    print ("training set score:{:.2f}".format(elastic.score(x_train,y_train)))
```

```
    print ("test set score:{:.2f}".format(elastic.score(x_test,y_test)))
    keyp =[]
    for i in range(len(coef)):
        if coef[i] != 0:
            keyp.append(columns[i])
    print(keyp)
    return LarsLasso
elasticnetcv()

#2-5
#RandomForestClassifier
def randomforest():
    clf = RandomForestClassifier()
    df = pd.read_excel('样本数据 z_normal.xlsx')
    x = df.values
    y = pd.read_excel('y.xlsx').values
    clf.fit(x, y.astype('int'))
    importance = clf.feature_importances_
    indices = np.argsort(importance)[::-1]
    features = df.columns
    for f in range(x.shape[1]):
        print(("%-*s %f" % (30, features[f], importance[indices[f]])))
randomforest()
```

问题三的 Python 代码

```
#第三问
import tensorflow as tf
import os
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_squared_error, mean_absolute_error
from sklearn.svm import SVR
import math
from sklearn.ensemble import RandomForestRegressor
from sklearn.datasets import make_regression

#3-1 SVR 模型
x = pd.read_excel('x.xlsx').values
y = pd.read_excel('x.xlsx').values
y=y.ravel()
x_train = x[:-81]
y_train = y[:-81]
```

```
x_test = x[-81:]
y_test = y[-81:]
clf = SVR(kernel='linear')
#clf = SVR(kernel='poly')
#clf = SVR(kernel='rbf')
clf.fit(x_train, y_train)
y_pred=clf.predict(x_test)
mse = mean_squared_error(y_pred,y_test)
rmse = math.sqrt(mean_squared_error(y_pred,y_test))
mae = mean_absolute_error(y_pred,y_test)
print('均方误差: %.6f' % mse)
print('均方根误差: %.6f' % rmse)
print('平均绝对误差: %.6f' % mae)
plt.plot(y_pred, color='orangered', label='Predict RONloss')
plt.plot(y_test, color='green', label='Real RONloss')
plt.title('RONloss Prediction')
plt.legend()
plt.show()

#3-2 BP 神经网络
x = pd.read_excel('x.xlsx').values
y = pd.read_excel('x.xlsx').values
x_train = x[:-81]
y_train = y[:-81]
x_test = x[-81:]
y_test = y[-81:]
model = tf.keras.models.Sequential([
    tf.keras.layers.Dense(14, activation='relu', kernel_regularizer=tf.keras.regularizers.l2()),
    #tf.keras.layers.Dropout(0.2),
    tf.keras.layers.Dense(10, activation='relu', kernel_regularizer=tf.keras.regularizers.l2()),
    tf.keras.layers.Dropout(0.2),
    tf.keras.layers.Dense(1, activation='relu', kernel_regularizer=tf.keras.regularizers.l2())
])
model.compile(optimizer=tf.keras.optimizers.Adam(lr=0.0003),
                loss='mse',
                metrics=['accuracy'])

model.fit(x_train, y_train, batch_size=32, epochs=500, validation_data=(x_test,y_test),
validation_freq=20)
model.summary()
y_pred=model.predict(x_test)
print(y_pred)
print(y_pred.shape)
print(y_test.shape)
```

```
mse = mean_squared_error(y_pred,y_test)
rmse = math.sqrt(mean_squared_error(y_pred,y_test))
mae = mean_absolute_error(y_pred,y_test)
print('均方误差: %.6f' % mse)
print('均方根误差: %.6f' % rmse)
print('平均绝对误差: %.6f' % mae)
plt.plot(y_pred, color='orangered', label='Predict RONloss')
plt.plot(y_test, color='green', label='Real RONloss')
plt.title('RONloss Prediction')
#plt.xlabel('Factors')
#plt.ylabel('RONloss')
plt.legend()
plt.show()

#3-3 KNN 回归
x = pd.read_excel('x.xlsx').values
y = pd.read_excel('y.xlsx').values
x_train = x[:-80]
y_train = y[:-80]
x_test = x[-80:]
y_test = y[-80:]
from sklearn.neighbors import KNeighborsRegressor
neigh = KNeighborsRegressor(n_neighbors=6)
neigh.fit(x_train, y_train)
y_pred=neigh.predict(x_test)
mse = mean_squared_error(y_pred,y_test)
rmse = math.sqrt(mean_squared_error(y_pred,y_test))
mae = mean_absolute_error(y_pred,y_test)
print('均方误差: %.6f' % mse)
print('均方根误差: %.6f' % rmse)
print('平均绝对误差: %.6f' % mae)
plt.plot(y_pred, color='orangered', label='Predict RONloss')
plt.plot(y_test, color='green', label='Real RONloss')
plt.title('RONloss Prediction')
plt.legend()
plt.show()

#3-4 随机森林回归
x = pd.read_excel('x.xlsx').values
y = pd.read_excel('\y.xlsx').values
y=y.ravel()
x_train = x[:-81]
y_train = y[:-81]
x_test = x[-81:]
```

```
y_test=y[-81:]
x, y = make_regression(n_features=18, n_informative=2,
                                random_state=0, shuffle=True)
regr = RandomForestRegressor(max_depth=4, random_state=0,
                                        n_estimators=100)
regr.fit(x_train, y_train)
y_pred=regr.predict(x_test)
mse = mean_squared_error(y_pred,y_test)
rmse = math.sqrt(mean_squared_error(y_pred,y_test))
mae = mean_absolute_error(y_pred,y_test)
print('均方误差: %.6f' % mse)
print('均方根误差: %.6f' % rmse)
print('平均绝对误差: %.6f' % mae)
plt.plot(y_pred, color='orangered', label='Predict RONloss')
plt.plot(y_test, color='green', label='Real RONloss')
plt.title('RONloss Prediction')
plt.legend()
plt.show()

#3-5 MLP 神经网络
x = pd.read_excel('x.xlsx').values
y = pd.read_excel('y.xlsx').values
y=y.ravel()
x_train = x[:-81]
y_train = y[:-81]
x_test = x[-81:]
y_test = y[-81:]
from sklearn.neural_network import MLPRegressor
mlp=MLPRegressor(activation='relu', alpha=0.0001, batch_size='auto',early_stopping=False,
shuffle=True, solver='adam', tol=0.0001,
        validation_fraction=0.2, verbose=False, warm_start=False)
mlp.fit(x_train,y_train)
y_pred = mlp.predict(x_test)
y_pred=tf.cast(y_pred,dtype=tf.float64)
y_test=tf.cast(y_test,dtype=tf.float64)
print(y_pred)
mse = mean_squared_error(y_pred,y_test)
rmse = math.sqrt(mean_squared_error(y_pred,y_test))
mae = mean_absolute_error(y_pred,y_test)
print('均方误差: %.6f' % mse)
print('均方根误差: %.6f' % rmse)
print('平均绝对误差: %.6f' % mae)
plt.plot(y_pred, color='red', label='Predict RONloss')
plt.plot(y_test, color='blue', label='Real RONloss')
```

```
plt.title('RONloss Prediction')
plt.legend()
plt.show()
```

问题四的 Python 代码

```
#第四问
import pandas as pd
from matplotlib import pyplot as plt
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_squared_error, mean_absolute_error
from matplotlib.colors import LinearSegmentedColormap
import math
from mpl_toolkits.mplot3d.axes3d import Axes3D
import numpy as np
import matplotlib.cm as cmx
import matplotlib
from sklearn.svm import SVR

#4-2
#最优操作选取
x = pd.read_excel('x.xlsx').values
y = pd.read_excel('y.xlsx').values
y=y.ravel()
x_train = x[:-81]
y_train = y[:-81]
x_test = x[-81:]
y_test = y[-81:]
clf = SVR(kernel='linear')
clf.fit(x_train, y_train)
df = pd.read_excel('xr.xlsx')
columns = list(df.columns)
error = 0
x133 = df.iloc[132, :].values
y = pd.read_excel('y.xlsx').values
for a in np.arange(2.35, 2.7, 0.1):
    for b in np.arange(0, 121, 10):
        for c in np.arange(2, 101, 10):
            for d in np.arange(3, 76, 4):
                for e in np.arange(320, 482, 20):
                    for f in np.arange(0.4, 0.9, 0.1):
                        df[columns[-6]] = [(a-2.3859664)/(2.607782-2.3859664) for i in
range(df.shape[0])]
                        df[columns[-5]] = [(b-0.3016754)/(100.81921-0.3016754) for i
in range(df.shape[0])]
```

```
                                df[columns[-4]] = [(c-2.8080836)/(83.222635-2.8080836) for i
in range(df.shape[0])]
                                df[columns[-3]] = [(d-3.6843995)/(64.396493-3.6843995) for i
in range(df.shape[0])]
                                df[columns[-2]] = [(e-334.99402)/(457.82386-334.99402) for i
in range(df.shape[0])]
                                df[columns[-1]] = [(f-0.4305181)/(0.6833051-0.4305181) for i
in range(df.shape[0])]
                                x1 = df.values
                                # print(x1.shape)
                                y_pred = clf.predict(x1)
                                # plt.plot(y_pred, color='red', label='Predict RONloss')
                                # plt.plot(y, color='blue', label='Real RONloss')
                                # plt.title('RONloss Prediction')
                                # plt.show()
                                if error < (np.mean((y-y_pred)/y)):
                                    params = (a, b, c, d, e, f)
                                    error = (np.mean((y-y_pred)/y))
                                    print('******', params, (np.mean((y-y_pred)/y)))
                                    print('******', y[132], '---->',
clf.predict(x133[np.newaxis, :]))
                print((a, b, c, d, e, f))
                print('==========')
print('*******')
print(params)

df = pd.read_excel('xr.xlsx')
columns = list(df.columns)
y = pd.read_excel('y.xlsx').values
a, b, c, d, e, f = params[0], params[1], params[2], params[3], params[4], params[5]
df[columns[-6]] = [(a-2.3859664)/(2.607782-2.3859664) for i in range(df.shape[0])]
df[columns[-5]] = [(b-0.3016754)/(100.81921-0.3016754) for i in range(df.shape[0])]
df[columns[-4]] = [(c-2.8080836)/(83.222635-2.8080836) for i in range(df.shape[0])]
df[columns[-3]] = [(d-3.6843995)/(64.396493-3.6843995) for i in range(df.shape[0])]
df[columns[-2]] = [(e-334.99402)/(457.82386-334.99402) for i in range(df.shape[0])]
df[columns[-1]] = [(f-0.4305181)/(0.6833051-0.4305181) for i in range(df.shape[0])]
x = df.values
y_pred = clf.predict(x)
print(np.mean((y-y_pred)/y))
print(y_pred[132])
plt.plot(y_pred, color='red', label='Predict')
plt.plot(y, color='blue', label='Real')
plt.legend(['Predict', 'Real'])
plt.title('RONloss Prediction')
```

```
plt.show()
```

问题五的 Python 代码

```
#第五问
import pandas as pd
from matplotlib import pyplot as plt
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_squared_error, mean_absolute_error
from matplotlib.colors import LinearSegmentedColormap
import math
from mpl_toolkits.mplot3d.axes3d import Axes3D
import numpy as np
import matplotlib.cm as cmx
import matplotlib
from sklearn.svm import SVR

x = pd.read_excel('x.xlsx').values
y = pd.read_excel('y.xlsx').values
y=y.ravel()
x_train = x[:-81]
y_train = y[:-81]
x_test = x[-81:]
y_test = y[-81:]
clf = SVR(kernel='linear')
clf.fit(x_train, y_train)
params = (2.65, 0, 92, 3, 480, 0.8)
delta = (0.1, 10, 1, 1, 1, 2, 0.1)
df = pd.read_excel('x133.xlsx')
x133 = pd.read_excel('x133_normal.xlsx').values
print(x133[0][7])
y133 = 1.31
columns = list(df.columns)
action_old = [float(df.values[:, i-6]) for i in range(6)]
max_step = 0
for i in range(6):
    step = int(abs(action_old[i] - params[i])/delta[i])
    if step > max_step:
        max_step = step
loss = []
s = []
t_action = action_old
imgX = []
for i in range(max_step):
    print('---{}---'.format(i))
```

```
    for j in range(6):
        if abs(t_action[j] - params[j]) > delta[j]:
            if (t_action[j] - params[j]) > delta[j]:
                t_action[j] -= delta[j]
            else:
                t_action[j] += delta[j]
    #print(t_action)
    normal_action = [0, 0, 0, 0, 0, 0]
    normal_action[-6] = (t_action[-6]-2.3859664)/(2.607782-2.3859664)
    normal_action[-5] = (t_action[-5]-0.3016754)/(100.81921-0.3016754)
    normal_action[-4] = (t_action[-4]-2.8080836)/(83.222635-2.8080836)
    normal_action[-3] = (t_action[-3]-3.6843995)/(64.396493-3.6843995)
    normal_action[-2] = (t_action[-2]-334.99402)/(457.82386-334.99402)
    normal_action[-1] = (t_action[-1]-0.4305181)/(0.6833051-0.4305181)
    for j in range(6):
        x133[0][j-6] = normal_action[j]
    x133[0][7] = (i/max_step) * (0.3333)
    s.append(x133[0][7] * 5.4 + 3.2)
    print(clf.predict(x133))
    loss.append(float(clf.predict(x133)))
    imgX.append([t_action[-4], t_action[-3], t_action[-2]])

plt.plot(s)
plt.title('133 产品硫含量随时间变化')
plt.grid()
plt.text(5, 3.2, (0, 3.2), color='r')
plt.plot(0, 3.2, '.', color='r')
plt.text(75, 5, (86, 5), color='r')
plt.plot(86, 5, '.', color='r')
plt.xlabel('Time step')
plt.ylabel('产品硫含量（μg/g）')

plt.show()
plt.plot(loss)
plt.title('133 RON 损失值随时间变化')
plt.grid()
plt.text(2, 1.275, (0, 1.28), color='r')
plt.text(77, 0.87, (86, 0.88), color='r')
plt.plot(0, 1.28, '.', color='r')
plt.plot(86, 0.885, '.', color='r')
plt.xlabel('Time step')
plt.ylabel('RON loss')
plt.show()
print(imgX)
```

```
def plot_embedding_3d_xs(x, y, title=None, c1=2, c2=2, c3=200):
    x_min, x_max = np.min(x, axis=0), np.max(x, axis=0)
    x = (x - x_min)/(x_max - x_min)
    fig = plt.figure()
    y_normal = (y-np.min(y))/(np.max(y)-np.min(y))
    ax = fig.add_subplot(1,1,1, projection='3d')
    cmap = cmx.get_cmap('rainbow', 20)
    for i in range(x.shape[0]):
        if i%10==0:
            if i==0:
                ax.text(x[i,0], x[i,1], x[i,2], 'start:'+str(round(y[i], 2)), color=cmap(1-
y_normal[i]), fontdict={'weight':'bold', 'size':14})
            elif abs(i-x.shape[0]) <= 10:
                pass
            else:
                ax.text(x[i,0], x[i,1], x[i,2], str(round(y[i], 2)), color=cmap(1-y_normal[i]),
fontdict={'weight':'bold', 'size':14})
        elif i == x.shape[0]-1:
            ax.text(x[i,0], x[i,1], x[i,2], 'end:'+str(round(y[i], 2)), color=cmap(1-y_normal[i]),
fontdict={'weight':'bold', 'size':14})
        ax.text(x[i,0], x[i,1], x[i,2], 'x', color=cmap(1-y_normal[i]), fontdict={'size':10})
    ax.set_xlabel('S-ZORB.TE_7508B.DACA')
    ax.set_ylabel('S-ZORB.TE_7108B.DACA')
    ax.set_zlabel('S-ZORB.TC_1607.DACA')
    if title is not None:
        plt.title(title)
        plt.show()

def plot_embedding_3d(x, y, title=None, c1=2, c2=2, c3=200):
    x_min, x_max = np.min(x, axis=0), np.max(x, axis=0)
    x = (x - x_min)/(x_max - x_min)
    fig = plt.figure()
    y_normal = (y-np.min(y))/(np.max(y)-np.min(y))
    ax = fig.add_subplot(1,1,1, projection='3d')
    cmap = cmx.get_cmap('rainbow', 20)
    for i in range(x.shape[0]):
        if i == 0:
            ax.text(x[i,0], x[i,1], x[i,2], str(round(y[i], 2))+'_start', color=cmap(1-
y_normal[i]), fontdict={'weight':'bold', 'size':14})
        elif i == x.shape[0]-1:
            ax.text(x[i,0], x[i,1], x[i,2], str(round(y[i], 2))+'_end', color=cmap(1-
y_normal[i]), fontdict={'weight':'bold', 'size':14})
        else:
```

```
                ax.text(x[i,0], x[i,1], x[i,2], str(round(y[i], 2)), color=cmap(1-y_normal[i]),
fontdict={'weight':'bold', 'size':14})
    ax.set_xlabel('S-ZORB.TE_7508B.DACA')
    ax.set_ylabel('S-ZORB.TE_7108B.DACA')
    ax.set_zlabel('S-ZORB.TC_1607.DACA')
    if title is not None:
        plt.title(title)
        plt.show()

plot_embedding_3d_xs(imgX, s, '133  产品硫含量与三个操作变量变化关系', 2, 2, 200)
plot_embedding_3d_xs(imgX, loss, '133 RON 损失与三个操作变量变化关系', 3, 3, 120)
plot_embedding_3d(imgX, s, '133  产品硫含量与三个操作变量', 2, 2, 200)
plot_embedding_3d(imgX, loss, '133 RON 损失与三个操作变量', 3, 3, 120)
```