基于 SVM 支撑向量分类机与 KMeans 聚类分析的古代玻璃成分分析模型

摘要

针对玻璃文物的成分分析问题，本文建立了 SVM 支撑向量机模型、KMeans 聚类分析模型，运用了卡方分析、独立样本 t 检验、包络线拟合、单因素方差分析等判断方法，提出了类型划分依据及亚类划分方案，并对多重指标与不同化学成分含量间的关系进行具体分析。

针对问题一，为分析风化程度与基本信息间的关系，本文使用卡方分析法。首先对数据进行清洗并归一化处理；之后对指标组合进行卡方分析，得到玻璃风化程度与玻璃类型之间的渐进显著性为 0.049，可以认为其具显著相关性，而风化程度与玻璃纹饰、颜色间并无显著相关性；最后结合统计柱状图得到可能与玻璃表面风化相关的组合，如（高钾，B，蓝绿）。

为分类分析玻璃风化程度与化学成分含量的关系，本文采用相关性检验和独立样本 t 检验方法。首先按分类样本绘制雷达图得到直观的成分含量；之后通过计算 Pearson 相关系数判断每类样本中不同化学成分含量之间的相关性，如高钾风化样品中 CaO 与 SiO_2,Al_2O_3 含量具有强负相关性；最后运用 SPSS 进行独立样本 t 检验，得到每种成分风化前后含量是否具有独立性，如高钾玻璃风化前后 SiO_2、K_2O、CaO、MgO、Al_2O_3、Fe_2O_3 含量不独立，差异显著。

为根据风化点检测数据预测未风化时各成分含量，根据独立样本 t 检验结果将化学成分分类分析，分别绘制风化前后差异明显的指标随 SiO_2 含量变动的样本点，运用包络线拟合样本集中分布区域边界，代入风化前 SiO_2 含量预测值实现多种成分的预测。

针对问题二，为分析玻璃类型及亚类分类规律，本文建立 SVM 支撑向量分类机模型与 KMeans 聚类分析模型。首先对于划分高钾铅钡类型，建立 SVM 支撑向量机模型，通过求解二次规划问题，确定最优超平面具体参数；其次对于每个类别进行亚类划分，建立 KMeans 聚类分析模型。通过绘制 “SSE-聚类个数曲线” 找到合适的聚类个数，分别对高钾、铅钡玻璃实现聚类划分，得到铅钡玻璃的 5 个亚类 $C_i\ (i = 1, 2, \cdots, 5)$ 与高钾玻璃的 3 个亚类 $D_i\ (i = 1, 2, 3)$；最后在模型的合理性和敏感性检验中引入了 Hausdorff 距离，克服了传统模型敏感性检验方法难以衡量点集轮廓的缺陷，验证了模型的稳健。

针对问题三，为确定未分类文物所属类型及亚类，本文将文物数据依次代入问题二中求解的 SVM 支撑向量分类机模型与 KMeans 聚类分析模型，得到具体分类结果为 A1、A6、A7 属于高钾玻璃，A2、A3、A4、A5、A8 属于铅钡玻璃，A1 属于 D2，A6、A7 属于 D0，A4、A8 属于 C4，A5 属于 C1。再对样本数据加入不同程度的 Gauss 噪声，依次代入两种模型，通过模型结果的准确性验证模型敏感性。

针对问题四，为分析不同类别玻璃样品间化学成分含量的关联与差异，本文建立单因素方差分析模型。对于玻璃的类型、颜色、纹饰类别，首先结合问题二中求解的高钾铅钡玻璃分类规律，可得类型类别的分析结果；其次对其他两类分别进行方差分析，根据显著性大小得到不同颜色玻璃文物间含量差异较大的化学成分为 CuO、PbO、SrO 等，进而推测这些成分含量会是不同玻璃颜色的成因；而不同纹饰玻璃文物间化学成分含量无明显差异。

综上，本文由数据驱动，运用 SVM 支撑向量机、KMeans 聚类分析等多种模型方法实现了古代玻璃的成分分析，对于古代玻璃文物鉴别分类具有良好的适用性。

关键词： 卡方分析，独立样本 t 检验，SVM 支撑向量机，KMeans 聚类分析，单因素方差分析

一、 问题背景与重述

1.1 问题背景

玻璃通过丝绸之路从西方传入我国，成为了早期贸易往来的宝贵物证。研究表明，中国最晚自战国晚期便已经开发出了独特的玻璃配方[1]，因此中国古代自制玻璃也具有与外来玻璃不同的化学成分。由于炼制过程中加入的助熔剂种类不同，玻璃成品中的主要化学成分往往也具有明显差异，继而我们可以把古代玻璃分为铅钡玻璃、钾钙玻璃、钠钙玻璃三类[2]。古代玻璃极易风化，这一过程中内外部元素大量交换往往导致其成分比例变化。研究风化导致的玻璃文物氧化物含量变动，不仅有助于考古工作者更准确地鉴别文物，也能对更好地保护玻璃文物带来启发。

1.2 问题重述

请根据所给的玻璃文物基本信息及化学成分比例数据建立模型，解决以下问题：

问题 1：根据附件表单 1 的相关信息分析玻璃文物的表面风化与其玻璃类型、纹饰和颜色的关系；按照玻璃类型、文物样品表面有无风化分类，研究每一类中化学成分含量的统计规律。并根据风化点检测数据，预测其风化前的化学成分含量。

问题 2：依据附件表单 2 数据分析高钾玻璃、铅钡玻璃的分类规律；对于每个类别选择合适的化学成分进行亚类划分，给出具体的划分方法及划分结果。再对分类结果的合理性和敏感性进行分析。

问题 3：对附件表单 3 中未分类玻璃文物的化学成分进行分析，鉴别其所属类型，并对分类结果的敏感性进行分析。

问题 4：针对不同类别的玻璃文物样品，分析其化学成分之间的关联关系，并比较不同类别之间的化学成分关联关系的差异性。

二、 问题分析

在解决问题之前将进行数据的预处理，根据题中“将成分比例累加和介于 85% 到 105% 之间的数据视为有效数据”条件，初步筛选去掉范围外的数据，再将表中的数值归一化，保证每件文物各种成分占比和为 1。

针对问题一第一部分，为确定玻璃文物表面风化与基本信息间的关系，考虑到样本的总体分布未知，故拟采用定类变量法中的卡方分析。分别绘制风化玻璃类型、纹饰、颜色与未风化玻璃对应指标的统计图，进而综合考量玻璃风化程度与基本信息的两两关系和组合关系。

针对问题一第二部分，为按类型分析风化前后各化学成分含量的统计规律，本文拟采用分类分析雷达图、相关性检验、独立样本 t 检验方法。首先为直观得到不同类型不同风化程度玻璃样品各成分含量，拟按分类样本分别绘制雷达图。通过观测雷达图，将得到每类样品化学元素含量的普遍规律。其次由于玻璃样品中化学元素种类丰富存在形式多样，猜测其化学组成成分存在某种相关性规律，故对其进行相关性检验，定量衡量相关性强弱。最后为了得到风化对高钾、铅钡玻璃内元素含量的影响，本文将分别对高钾、铅钡玻璃进行独立样本 t 检验，进而得到在不同显著性水平下呈现差异的化学成分。

针对问题一第三部分，为根据风化后样本化学成分含量预测风化前对应含量，拟对铅钡玻璃、高钾玻璃样本点集中分布区域分别包络拟合预测。重点考虑此前独立样本 t

检验得出风化前后含量差异大的化学成分，求解这些成分随 SiO_2 含量变化包络线，根据风化前 SiO_2 含量的预测值，实现这些成分预测含量的求解。

针对问题二第一部分，为找出玻璃类型的分类规律，将玻璃样品看作化学成分指标空间上的点集，拟找出一种空间内的分割，使得高钾、铅钡玻璃能够被分割完全隔开，代入相关数据判断以超平面作为分割的合理性与可行性，若可行，求解得出具体模型结果。

针对问题二第二部分，为对已按玻璃类型分类的样品进行亚类划分，使得亚类中样本性质相近，考虑采取 KMeans 聚类分析法，结合计算 SSE 随分簇数的变化趋势，确定合适的分簇数，对样本点进行聚类分析，可将位于同一聚类的元素归为同一亚类。

针对问题二第三部分，为检验亚类划分模型的合理性与敏感性，考虑到样本数目有限，拟在传统方法上加以改进，将引入 Hausdoff 距离作为衡量聚类偏移量的指标。当样本噪声大小保持在一定范围内时，若原聚类间豪斯多夫距离的最小值大于加入噪声后聚类偏移量之和，即可保证模型的稳健性。

针对问题三，为实现未分类样品的归类，拟将样品数据依次代入此前的 SVM 支撑向量机模型和 KMeans 聚类分析模型，找到样品所属玻璃类型与亚类。为实现分类结果的敏感性分析，对样品数据加入 Gauss 噪声，以与分类结果的相似度为正确率的评判标准，绘制“正确率——噪声大小”曲线，即可找到模型高正确率对应的噪声范围，继而在此范围内分类模型稳健。

针对问题四，为分析不同类别玻璃文物样品化学成分含量间的关联与差异，首先按玻璃类型、颜色、纹饰分类，拟对颜色、纹饰化学成分含量间的关联与差异采用方差分析法，其次对玻璃类型化学成分含量间的关联与差异，问题二中的 SVM 支撑向量机模型已给出答案。最后还应结合相关文献，总结关联及差异。

三、 模型假设

1) 假设文物未检测出的化学成分近似为 0。

2) 假设除了附件 2 中列出的化学成分之外，玻璃文物不包含其他的化学成分。

3) 假设不同玻璃文物化学成分含量之间没有关联。

四、 符号说明

符号	含义
O_i	第 i 类玻璃的观测频数
E_i	第 i 类玻璃的期望频数
m_0	每一类玻璃的样本数
$\boldsymbol{a}=(a_1,\cdots,a_{14})$	每一类中的玻璃样本
a_j	玻璃样本的各种化学成分含量
a_{jl}	第 l 个样本的第 j 个化学成分含量
$\bar{a}_j$	第 j 个化学成分含量的样本均值
$\boldsymbol{a}_i$	每一类的第 i 个玻璃样本
D	玻璃文物颜色种类
D_j	玻璃文物的第 j 个颜色种类
n_j	种类 D_j 下的数据量
E_{ij}	第 i 个种类第 j 个文物化学成分含量

五、 问题一：指标关系、风化影响及含量预测

可根据题意将问题一分为三个小问。第一小问通过卡方分析、绘制统计图表，找出玻璃文物表面风化与类型、纹饰、颜色之间的两两关系和组合关系。第二小问通过相关性检验与独立样本 t 检验，找出影响（高钾，风化）（高钾，未风化）（铅钡，风化）（铅钡，未风化）这四种类型玻璃文物的化学成分含量规律。第三小问在独立样本 t 检验结果的基础上，对样本化学成分进行分类预测。对铅钡玻璃、高钾玻璃样本点集中分布区域分别包络拟合预测，实现这些成分预测含量的求解。

5.1 风化程度与各项基本指标间关系

为分析玻璃文物的表面风化与玻璃类型、纹饰、颜色间的关系，首先将风化程度与这三项基本指标分别组合，通过卡方分析得到对应的渐进显著性大小，进而说明组合内指标间的相关性。其次由于玻璃文物类型，纹饰，颜色之间存在一定的相关性，可以分别绘制在表面风化和表面未风化情况下其他各项指标样本的统计图，说明。最后统计得到占比最高的指标组合与文物表面风化程度具有较强的相关性。

5.1.1 数据的预处理

首先根据题中“将成分比例累加和介于 85% 到 105% 之间的数据视为有效数据”这一信息，对附件表单 2 的数据进行清洗。不难发现 15，17 号玻璃文物成分比例累加分别为 79.47%、71.89% 不符合要求，故在数据集中删去其相关数据，在后续问题的解答中也不再使用相关信息。由于受到检测手段等限制，各成分比例累加和不一定为 1，故不妨将所有成分占比归一化处理。

5.1.2 基于卡方分析的两两相关性判断

为分析风化情况与玻璃类型，风化情况与玻璃纹饰，风化情况与玻璃颜色之间的两两关系，考虑到样本总体分布未知且基本信息间无偏序关系，不宜简单的赋值量化，故采用卡方分析的统计学原理求解其相关性。以风化情况与玻璃类型为例，此时可将风化、未风化与铅钡玻璃、高钾玻璃两两组合共得到 4 种分类情况 $(m = 4)$，将观测频数记为 $O_i(i = 1, 2, \cdots, m)$。再假设这两项指标相互独立，可推出此时的期望频数 E_i。

那么 Pearson 卡方 χ^2 为

$$\chi^2 = \sum_{i=1}^{m} \frac{(O_i - E_i)^2}{E_i}, \tag{1}$$

代入附件表单 1 相关数据，计算可得：

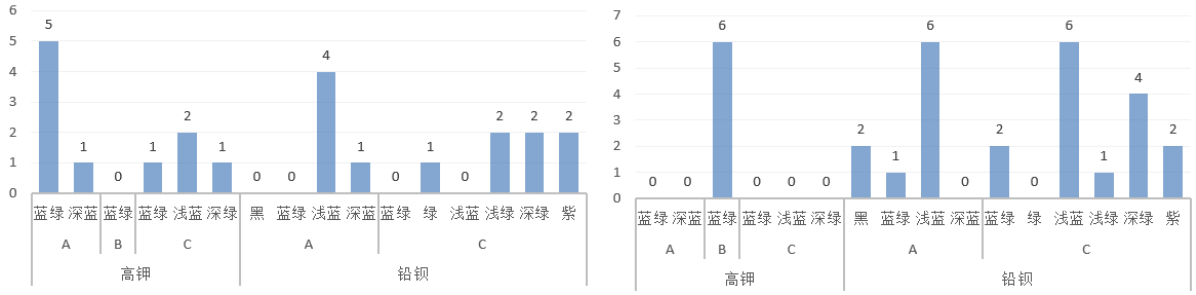
玻璃文物表面风化程度与的玻璃类型之间的渐进显著性为 0.049<0.05，因此这两项指标之间具有显著相关性。

玻璃文物表面风化程度与的玻璃纹饰之间的渐进显著性为 0.057>0.05，这表明接受假设“这两项指标不相关”，同时拒绝备择假设“这两项指标显著相关”。

玻璃文物表面风化程度与的玻璃颜色之间的渐进显著性为 0.428>0.05，这表明这两项指标之间不具有显著相关性。

5.1.3 基于统计图的指标组合

其次考虑到玻璃类型、纹饰、颜色的不同组合对应的样本数目具有显著差异，例如符合（铅钡，A，浅蓝）组合有 10 种，而（铅钡，C，绿）仅有一组，故推测这三项指标之间具有一定的相关性。因而分别统计表面风化和未分化两种玻璃文物的相关信息，并绘制统计图如图1a、图1b。



(a) 未风化分布图　　**(b) 风化分布图**

图 1　玻璃各特征的分布

由图一知，在表面风化的玻璃文物中，高钾玻璃仅有（高钾，B，蓝绿）这一种，铅钡玻璃则多以（铅钡，A，浅蓝）（铅钡，C，浅蓝）（铅钡，C，深绿）的组合形式出现，可以推测玻璃文物表面风化与三项指标的上述四种组合相关。

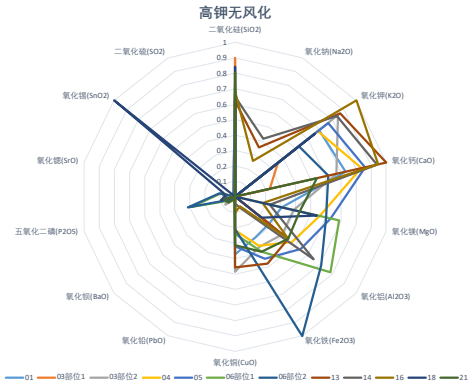
由图二知，在表面未风化的玻璃文物中，（高钾，A，蓝绿）（铅钡，A，浅蓝）是主要的组合形式，故可以推测玻璃文物表面未风化与这两种组合相关。

5.2 依类型分析风化前后化学成分含量规律

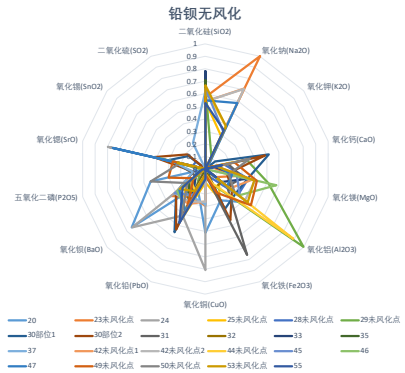
为按玻璃类型分析玻璃表面有无风化化学成分含量的统计规律，首先将样本分为高钾风化、高钾未风化、铅钡风化、铅钡未风化四种，分别绘制各成分含量占比雷达图，得到直观的统计规律。其次可以通过计算 Person 相关系数分别对高钾样本风化前后数据和铅钡样本风化前后数据分别进行相关性检验。最后再进行独立样本 t 检验，找出不同类型玻璃样本风化前后化学含量变化显著的成分。

5.2.1 各化学成分含量雷达图

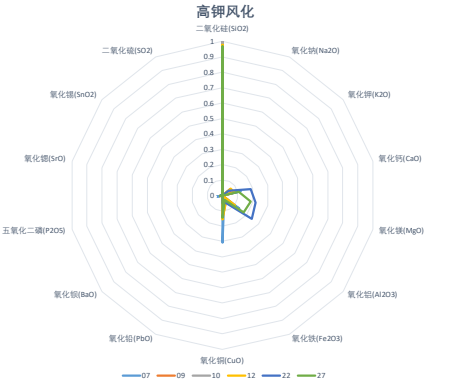
为了直观表示样本各化学成分的含量，利用按玻璃类型、玻璃表面有无风化分类的 4 种样本数据分别绘制各化学成分含量雷达图如图2



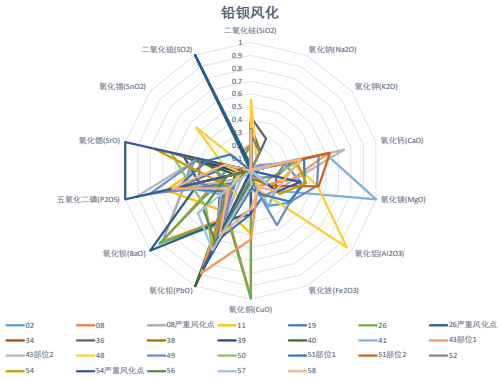
(a) 高钾无风化玻璃各化学成分雷达图



(b) 铅钡无风化玻璃各化学成分雷达图



(c) 高钾风化玻璃各化学成分雷达图



(d) 铅钡风化玻璃各化学成分雷达图

图 2　雷达图

在高钾风化样本中，样本 SiO_2 含量占比普遍较高，MgO，CuO 含量极低。

在高钾未风化样本中，样本 CaO，K_2O 含量最高，SiO_2，Fe_2O_3，Al_2O_3 含量也相对较高。

在铅钡无风化样本中，样本 PbO，CaO，Fe_2O_3 含量较高，其余组分含量相差较大，但普遍具有少量 BaO。

在铅钡风化样本中，样本 PbO，BaO，P_2O_5 含量最高，SrO，CaO，MgO 含量相对较高。

5.2.2 相关性检验

考虑到样本中 SiO_2 占比较高，且在高钾玻璃风化过程中随着表面富硅层的形成 SiO_2 含量逐渐上升、在铅钡玻璃风化过程中 SiO_2 侵蚀流失严重含量逐渐下降[3]，由于各种元素占比和在进行归一化后均为一，故推测不同类型样品中部分氧化物与 SiO_2 含量呈负相关性或正相关性。

考虑用 Pearson 相关性检验分析高钾无风化、高钾风化、铅钡无风化、铅钡风化这四种样本中每种化学成分的具体相关性强弱。不妨以高钾无风化样本为例，用 $\boldsymbol{a} = (a_1, a_2, \cdots, a_{14})$ 表示第 i 件高钾无风化玻璃文物，其中 $a_j\ (j = 1, \cdots, 14)$ 按 j 递增依次为 $SiO_2, \cdots, SO_2$ 等化学成分指标。那么对于其中两个不同的成分 $a_j, a_k\ (j \neq k)$，它们的 Pearson 相关系数为

$$r_{jk} = \frac{\sum_{l=1}^{m_0}(a_{jl} - \bar{a}_j)(a_{kl} - \bar{a}_k)}{\left(\sum_{l=1}^{m_0}(a_{jl} - \bar{a}_j)^2\right)^{1/2}\left(\sum_{l=1}^{m_0}(a_{kl} - \bar{a}_k)^2\right)^{1/2}}, \tag{2}$$

其中 a_{jl} 表示第 l 个样本的第 j 项化学成分，$\bar{a}_j$，$\bar{a}_k$ 是这些成分的样本均值。

计算得到的 Pearson 相关系数可以表征两种成分的相关性，部分数据如表1：

表 1　Pearson 相关性分析部分表

		二氧化硅 (SiO2)	氧化钠 (Na2O)	氧化钾 (K2O)	氧化钙 (CaO)	氧化镁 (MgO)	氧化铝 (Al2O3)
二氧化硅 (SiO2)	皮尔逊相关性	1	-0.431	-.606*	-.669*	-0.142	-.594*
	显著性（双尾）		0.162	0.037	0.017	0.660	0.042
	平方和与叉积	0.089	-0.068	-0.162	-0.236	-0.033	-0.103
	协方差	0.008	-0.006	-0.015	-0.021	-0.003	-0.009
	个案数	12	12	12	12	12	12
氧化钠 (Na2O)	皮尔逊相关性	-0.431	1	0.531	0.575	-.596*	0.180
	显著性（双尾）	0.162		0.075	0.051	0.041	0.576
	平方和与叉积	-0.068	0.278	0.252	0.359	-0.248	0.055
	协方差	-0.006	0.025	0.023	0.033	-0.023	0.005
	个案数	12	12	12	12	12	12

完整表见支撑材料’ 高钾风化-皮尔逊输出.spv’，’ 铅钡风化-皮尔逊输出.spv” 高钾无风化-皮尔逊输出.spv” 铅钡无风化-皮尔逊输出.spv’。

根据判别准则，当显著性小于 0.05 时，说明两种成分相关性较强；当显著性大于 0.05，说明两种成分相关性不强。当相关性较强时，Pearson 相关系数大于 0，说明正相关性较强；小于 0 时，说明负相关性较强。

观察可得：（这里只集中描述在 0.01 级别下极强的相关关系）

在高钾风化玻璃样品中，CaO 与 SiO_2,Al_2O_3 含量具有极强的负相关性。

在铅钡无风化玻璃样品中，SiO_2 与 PbO,BaO 含量具有极强的负相关性，PbO、CaO 与 SnO_2 含量具有极强的正相关性，CuO 与 BaO 含量具有极强的正相关性。

在铅钡风化玻璃样品中，SiO_2 与 SrO，SO_2 含量具有极强的负相关性，SiO_2 与 Al_2O_3，SnO_2 含量具有极强的正相关性；CaO 与 MgO、Fe_2O_3 含量具有极强的正相关性，MgO 与 Al_2O_3、Fe_2O_3 含量具有极强的正相关性；MgO 与 BaO 有极强的负相关性；Al_2O_3 与 Fe_2O_3、SnO_2 含量具有极强的正相关性；BaO 与 SO_2 含量具有极强的正相关性。

5.2.3 独立样本 t 检验

根据类型将玻璃文物样品分为高钾和铅钡两类，为针对每类玻璃分析文物风化前后化学成分含量这两组定量数据间的是否具有独立性，考虑使用 SPSS 进行独立样本 t 检验。

首先进行莱文方差等同性检验判断数据是否具有方差齐性，再根据结果分别进行平均值等同性 t 检验。部分检验结果如表2，完整表见支撑材料’ 铅钡 t 检验-输出.spv’,’ 高钾 t 检验-输出.spv’

表 2　独立 t 检验结果部份表

		莱文方差等同性检验		平均值等同性 t 检验			
		F	显著性	t	自由度	显著性	
						单侧 P	双侧 P
二氧化硅	假定等方差	6.752	0.019	-6.900	16	0.000	0.000
	不假定等方差			-9.641	12.540	0.000	0.000
氧化纳	假定等方差	13.490	0.002	1.303	16	0.105	0.211
	不假定等方差			1.872	11.000	0.044	0.088
氧化钾	假定等方差	5.872	0.028	5.393	16	0.000	0.000
	不假定等方差			7.672	11.539	0.000	0.000
氧化钙	假定等方差	8.184	0.011	3.464	16	0.002	0.003
	不假定等方差			4.886	12.012	0.000	0.000
氧化镁	假定等方差	4.276	0.055	3.020	16	0.004	0.008
	不假定等方差			3.833	15.937	0.001	0.001

根据检验标准，高钾玻璃风化前后 SiO_2、K_2O、CaO、MgO、Al_2O_3、Fe_2O_3 含量在 0.01 显著性水平下呈现明显差异，这说明了风化会对高钾玻璃中这些化学成分含量产生较大影响。

铅钡玻璃风化前后 SiO_2、Na_2O、CaO、PbO、P_2O_5 含量在 0.01 显著性水平下呈现明显差异，SrO 含量在 0.05 显著性水平下呈现明显差异，这说明了风化会对铅钡玻璃中这些化学成分含量产生较大影响。

5.3 预测风化点风化前各化学成分含量

由于不同类型玻璃文物样本之间元素变化趋势存在较大差异，故考虑分别对高钾、铅钡玻璃进行风化前化学成分含量预测。

以铅钡玻璃为例，通过5.2.3中独立样本 t 检验，发现风化会对铅钡玻璃中 SiO_2、Na_2O、CaO、PbO、P_2O_5、SrO 含量产生较大影响；对 K_2O、MgO、Al_2O_3、Fe_2O_3、CuO、BaO、SnO_2 的含量影响不大，因此可以忽略风化对这些化学成分含量造成的影响。对分

化后含量基本为 0 的 Na_2O 无法采用拟合预测，故取风化前平均值 0.927212 作为对其风化前含量的预测结果。对仅在严重风化情况下才会出现的 SO_2，预测其风化前含量为 0。

下面只需对玻璃文物风化前 SiO_2、CaO、PbO、P_2O_5、SrO 含量进行预测。在5.2.3中的独立 t 检验表明，SiO_2 的显著性水平达 10^{-14}，说明在风化前后 SiO_2 的变化及其显著。故可以用 SiO_2 含量作自变量，绘制“其他元素——二氧化硅”的样本散点分布图，如图3和4。

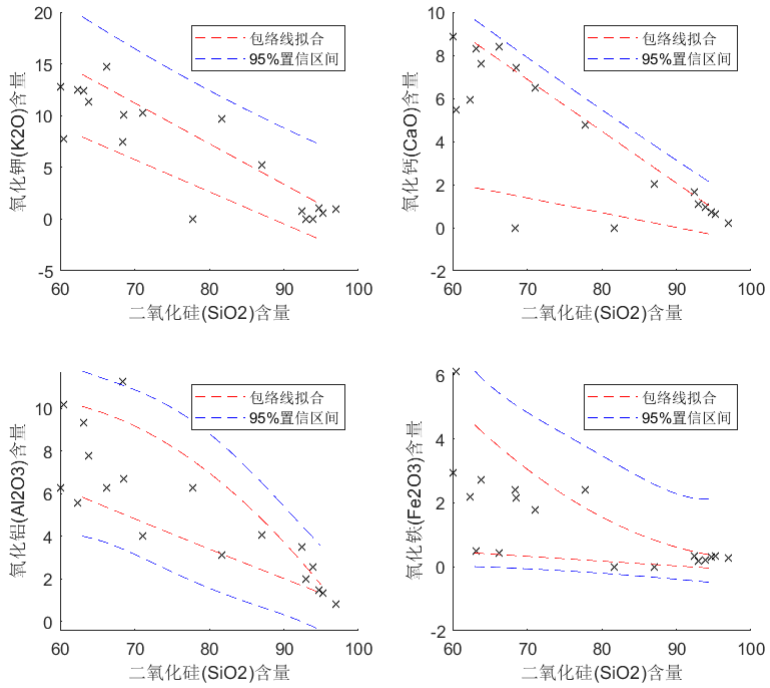


图 3　高钾类玻璃包络线拟合

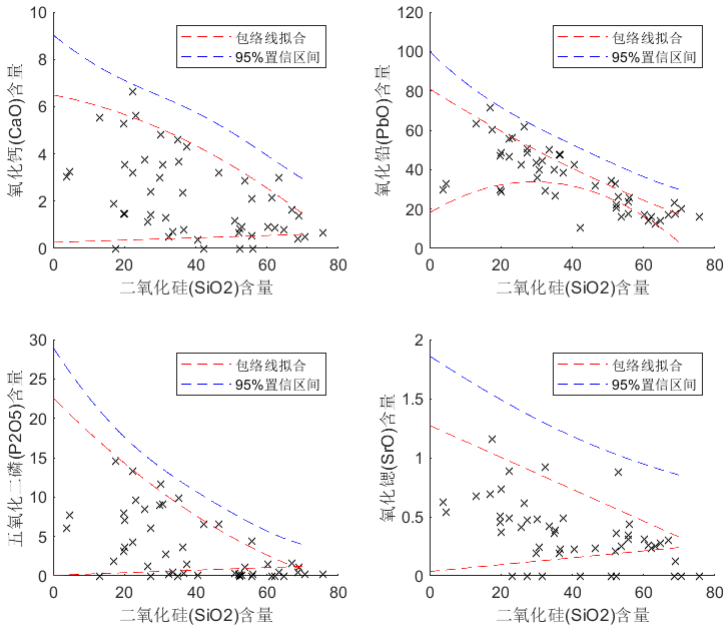


图 4　铅钡类玻璃包络线拟合

观察发现，样本点集中分布在图中的特定区域，但无明显函数规律，故考虑用两条包络线对分布区域的上下边界分别进行拟合。以 CaO 为例，将对分布边界拟合得到包络线表达式记为 $f_i(x)(f_i(x) = a_n x^n + a_{n-1}x^{n-1} + \cdots + a_1 x + a_0$，当 i=0 时表示上界拟合函数，当 i=1 时表示下界拟合函数)。

得到结果如下：

表 3　铅钡类玻璃文物

氧化钙	$f_0(x) = -0.00061x^2 - 0.02916x + 6.493919$
	$f_1(x) = 0.004909x + 0.25342$
氧化铅	$f_0(x) = 0.003257x^2 - 1.13898x + 81.04876$
	$f_1(X) = -0.01846x^2 + 1.079242 + 18.04178$
五氧化二磷	$f_0(x) = 0.002049x^2 - 0.45591x + 22.62081$
	$f_1(x) = 0.016274x + 0.081324$
氧化锶	$f_0(x) = -0.01349x + 1.273419$
	$f_1(X) = 0.002906x + 0.037155$

表 4　铅钡类玻璃文物

氧化钙	$f_0(x) = -0.00061x^2 - 0.02916x + 6.493919$
	$f_1(x) = 0.004909x + 0.25342$
氧化铅	$f_0(x) = 0.003257x^2 - 1.13898x + 81.04876$
	$f_1(X) = -0.01846x^2 + 1.079242 + 18.04178$
五氧化二磷	$f_0(x) = 0.002049x^2 - 0.45591x + 22.62081$
	$f_1(x) = 0.016274x + 0.081324$
氧化锶	$f_0(x) = -0.01349x + 1.273419$
	$f_1(X) = 0.002906x + 0.037155$

针对风化后样本点 (x_0, y_0) 其中 x_0 表示 SiO_2 含量，y_0 表示 CaO 含量。记 $g(x) = f(x) + b$，求解 $g(x_0) = y_0$ 得到 b，就针对这一样本点求出了化学成分含量拟合曲线。

通过相关文献可知未风化铅钡玻璃文物中 SiO_2 含量约为 $47.96\% \sim 72.61\%$，综合考量附件表单 2 中未风化铅钡玻璃 SiO_2 数据，认为未风化铅钡玻璃 SiO_2 含量为 $50\% \sim 70\%$，将风化后每个样本点数据依次代入拟合曲线，得出铅钡玻璃样品风化前 CaO 含量预测值的上下界。

例如计算可得

样品 41 的 CaO 含量介于 [3.121359012,5.44168047],

样品 50 的 CaO 含量介于 [1.37905206,3.685334155]。

同理，对已风化铅钡玻璃和高钾玻璃其他化学成分含量分别拟合，得到风化前各化学成分预测结果。全部数据见支撑材料' 高钾类玻璃未风化化学成分预测结果.xlsx'，' 铅钡类玻璃未风化化学成分预测结果.xlsx'

六、 问题二：类型、亚类划分与敏感性分析

由题意，问题二由三部分组成。第一部分为了找出高钾、铅钡玻璃的分类规律，采用 SVM 支撑向量分类机进行划分；第二部分为对每个类别的分类样品进行亚类划分，建立 KMeans 聚类分析模型，将化学成分含量相近的样品归为同一聚类，构成亚类划分；第三部分中，为检验分类方式合理性与敏感性，在原有数据集的基础上加入随机分布的高斯噪声并重新进行分类，比较前后分类结果的差别以说明模型的合理性和稳定性，并最后证明了模型的稳健。

6.1 SVM 分类机分析高钾、铅钡玻璃的分类规律

古代玻璃分为高钾、铅钡玻璃的分类方式，只和玻璃的化学组成成分相关，而与玻璃的颜色、纹饰无关。由于已有高钾、铅钡玻璃分类标签，对于该有标签的数据集，非常适合采用 SVM 支撑向量分类机，采用有监督的学习方式，以找出在玻璃的化学组成成分 $\mathbb{R}^{14}$ 空间上，将玻璃划分为高钾和铅钡玻璃的分界超平面。

本文建立 SVM 支撑向量分类机模型如下。将铅钡玻璃、高钾玻璃的每种化学成分都看作一个维度，将这些样本点看作 $\mathbb{R}^{14}$ 空间上的坐标，记为 $\boldsymbol{a}_i$(i=1,2,$\cdots$,67 将同一

文物的两个取样点对应的化学成分视作两个样本)。为找到 $\mathbb{R}^{14}$ 空间上的超平面，将整个空间分成两部分，保证高钾玻璃和铅钡玻璃分别位于这两个不同的部分中。对高钾玻璃标签，以代号 0 表示；对铅钡玻璃标签，以代号 1 表示。

由于数据量较少，为了避免过拟合现象的发生，将样本数据划分为训练集和测试集两个部分。用训练集训练 SVM 支撑向量分类机，测试集测试该分类器的表现。如果该分类器在测试集上表现良好，则能说明模型分类有效且未出现过拟合。

首先随机抽取 80% 的样本数据作为训练集，将目标划分的超平面记为

$$H = \left\{ \boldsymbol{x} \in \mathbb{R}^{14} | (\boldsymbol{\omega} \cdot \boldsymbol{x}) + b = 0 \right\}$$

其中 $\boldsymbol{\omega}$，b 为参数。分类需要保证铅钡玻璃对应的训练样本点集 ω_1 均在超平面上部, 高钾玻璃对应的训练样本点集 ω_2 均在超平面下部, 此外再保证超平面到两个点集的最近距离都尽可能地远，即最大化普通支持向量间的间隔 $\frac{2}{\|\boldsymbol{\omega}\|}$，就有

$$\begin{array}{ll} \max & \frac{2}{\|\boldsymbol{\omega}\|} \\ \text{s.t.} & \begin{cases} (\boldsymbol{\omega} \cdot \boldsymbol{a}_i) + b \geqslant y_i, & y_i = 1, \quad i \in \omega_1 \\ (\boldsymbol{\omega} \cdot \boldsymbol{a}_i) + b \leqslant y_i, & y_i = -1, \quad i \in \omega_2 \end{cases} \end{array}$$

那么寻找最优超平面的问题可转化为求解二次规划问题[4]。

$$\begin{array}{ll} \min & \frac{1}{2}\|\boldsymbol{\omega}\|^2, \\ \text{s.t.} & y_i[(\boldsymbol{\omega} \cdot \boldsymbol{a}_i) + b] \geqslant 1, \quad i = 1, \cdots, 67 \end{array}$$

为求解这一问题，引入 Lagrange 函数

$$L(\boldsymbol{\omega}, \boldsymbol{b}, \boldsymbol{\alpha}) = \frac{1}{2}\|\boldsymbol{\omega}\|^2 + \sum_{i=1}^{l} \alpha_i \{1 - y_i[\boldsymbol{\omega} \cdot \boldsymbol{a}_i + b]\},$$

式中: $\boldsymbol{\alpha} = [\alpha_1 \cdots, \alpha_{67}]^{\mathrm{T}}$ 为 Lagrange 乘子。原问题可转化为求解最优化问题（****）

$$\begin{array}{ll} \max_{\boldsymbol{\alpha}} & -\frac{1}{2}\sum_{i=1}^{67}\sum_{j=1}^{67} y_i y_j \alpha_i \alpha_j (\boldsymbol{a}_i \cdot \boldsymbol{a}_j) + \sum_{i=1}^{67} \alpha_i, \\ \text{s.t.} & \begin{cases} \sum_{i=1}^{67} y_i \alpha_i = 0 \\ \alpha_i \geqslant 0, i = 1, \cdots, 67 \end{cases} \end{array}$$

代入训练样本点数据，可得

$\boldsymbol{\omega} = (-0.02956241, 0.00551535, -0.05283312, -0.01644926, -0.00040758, 0.03743216, -0.02505811, -0.01378527, 0.10249466, 0.01348478, -0.02241906, 0.00158788, 0, 0)$

$b = 1.1452272$

将剩下的 20% 样本点数据代入超平面进行验证，得到准确为 1.0，即所有的测试集数据均被正确的划分。这就说明了 SVM 支撑向量机方法的可行性，表明通过超平面，可以正确的依据 $\mathbb{R}^{14}$ 空间上的玻璃化学组成成分，将其划分为高钾和铅钡玻璃。

由前所述，高钾玻璃的标签表示为 0，铅钡玻璃的标签表示为 1，则对于一化学成分组成已知的点，将其化学组成成分 $\boldsymbol{x}$ 代入超平面方程 $(\boldsymbol{\omega} \cdot \boldsymbol{x}) + b$: 若该点位于超平面之上 $((\boldsymbol{\omega} \cdot \boldsymbol{x}) + b > 0)$，则为铅钡玻璃；反之若该点位于超平面之下 $((\boldsymbol{\omega} \cdot \boldsymbol{x}) + b < 0)$, 则为高钾玻璃。因此，对于该表现有效的 SVM 支撑向量机，可以通过其超平面的参数方程 $(\boldsymbol{\omega} \cdot \boldsymbol{x}) + b = 0$($\boldsymbol{\omega}$,$b$ 如上所示)，分析得到高钾玻璃、铅钡玻璃的分类规律。

$\boldsymbol{\omega}$ 中绝对值较大的分量为 $\omega_3, \omega_6, \omega_9$, 对应的化学组分为氧化钾、氧化铝、氧化铅，其中 ω_3 的值为负，说明样品含有的氧化钾较高时，倾向被分类为高钾玻璃；而 ω_6, ω_9 的值为正，说明样品含有的氧化铝和氧化铅含量较高时，倾向于被分类为铅钡玻璃。

$\boldsymbol{\omega}$ 中绝对值较小的分量为 $\omega_2, \omega_5, \omega_{12}, \omega_{13}, \omega_{14}$, 对应的化学组分为氧化钠、氧化镁、氧化锶、氧化锡、二氧化硫。说明样品中含有的该组分含量，对玻璃划分为高钾玻璃和铅钡玻璃的影响性较小。

6.2 KMeans 聚类分析划分玻璃亚类

为了依据化学成分实现对高钾玻璃、铅钡玻璃的亚类划分，首先需要进行化学成分的合理选取。相关文献表明: 高钾玻璃在风化过程中，碱金属元素组分变动显著，如 Na,K 元素流失严重，而随着风化表面形成富硅层，Si 元素占比明显上升。铅钡玻璃在风化过程中，由于发生了水解氢化反应，Pb，Si 含量明显增加，Ba 含量显著降低。[3]此外在实际选取指标时，为了得到层次清晰类别显著的聚类划分结果，还需要确保样本对应的指标方差大小适中, 如果选取指标对应的方差太小，则说明在该类别中该指标的变动范围不大，不适应作为相关指标。

综合考虑上述因素，我们以 Si，K 作为高钾玻璃的划分指标，以 Pb,Ba,Si 作为铅钡玻璃的划分指标。

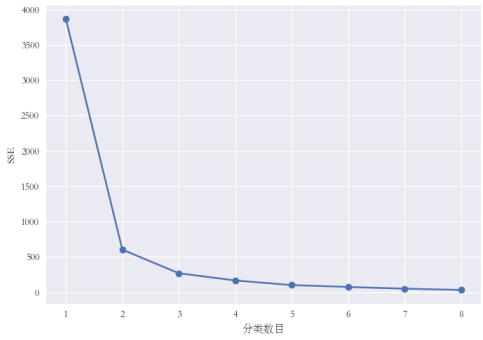
对于同一类型的玻璃，其亚类应有化学成分相近的特点，不同亚类之间化学组分应有明显差距。表现在由其各化学成分构成的 $\mathbb{R}^n$ 空间上，存在点集集中分布的趋势。故采取 KMeans 聚类分析法来找出点集的亚类。

为了找出最适合的聚类数目，采用各个点到聚类中心的距离平方和 SSE 来衡量

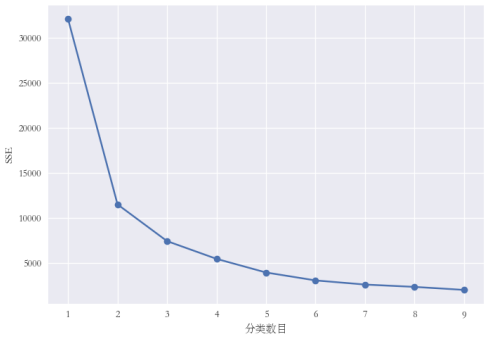
$$SSE = \sum_{C_j}^{m} \sum_{\boldsymbol{x}_j^{(i)} \in C_j} ||\boldsymbol{x}_j^{(i)} - \mu^{(j)}||_2^2$$

$C_1, C_2, ..., C_m$ 为聚类结果，为各个聚类的点集

首先绘制样本点簇内误差平方和 (SSE) 与聚类个数的图像，如图5a和5b:



(a) 高钾类玻璃样本 SSE



(b) 铅钡类玻璃样本 SSE

图 5　SSE 与分类数目的关系

观察发现，SSE 存在迅速下降过拐点后继而缓慢下降下降的变化趋势。在 SSE 迅速下降的时，说明各个点离其聚类中心的距离在迅速变小，模型的分类个数增加是有效的。在 SSE 下降缓慢时，说明此时增加分类数已经不能有效减小各个点离其聚类中心

的距离，模型出现过拟合。故采取下降趋势的拐点作为应当分类的亚类个数。即铅钡玻璃样本分成 5 个聚类、高钾玻璃样本分成 3 个聚类。如图6和7：

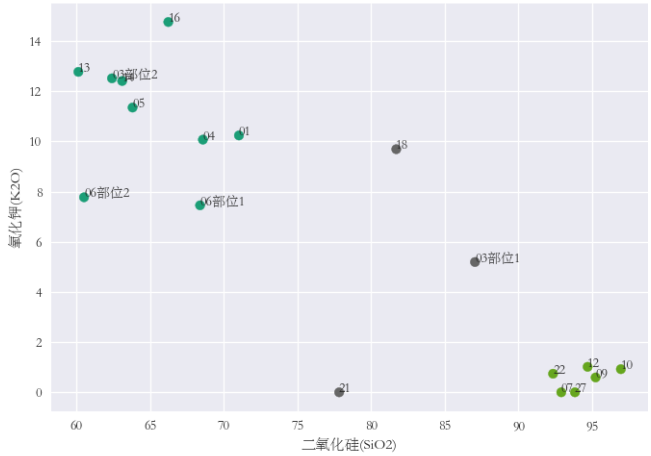


图 6　高钾类玻璃聚类分析结果

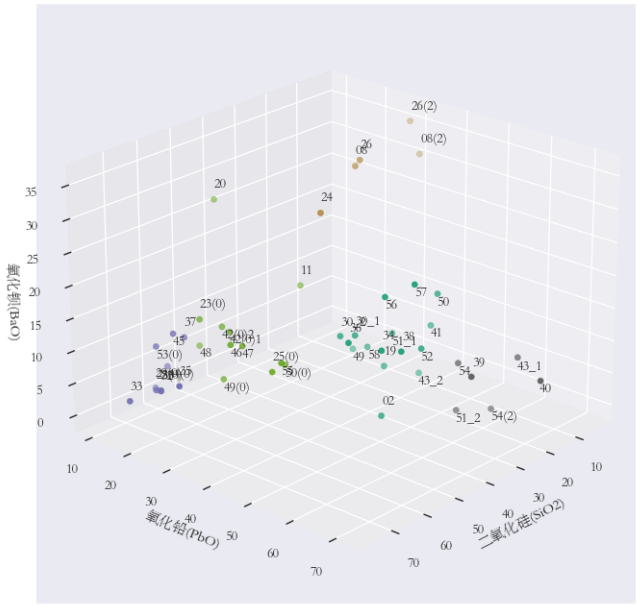


图 7　铅钡类玻璃聚类分析结果

得到聚类划分结果如图6和7。为了简化图形标注，对铅钡样本聚类中文物采样点标注进行简写（例如：将 30 号样本部位 1，2 分别记为 “30_1，30_2”，将 53 号样本未风化点记为 53(0)，将 54 号样本严重风化点记为 54(2)）。

由聚类结果，R_3 空间中铅钡玻璃样本点分为 5 个点集，用不同颜色标注，记为 C_i $(i = 1, 2, \cdots, 5)$，将这 5 个不同聚类作为铅钡玻璃的亚类。同理，高钾玻璃样本点分为

R_2 中的 3 个点集，记为 $D_i\,(i=1,2,3)$，将这 3 个不同聚类作为高钾玻璃的亚类。划分结果如表5和6所示，只展示了部分数据，全部数据见支撑文件' 高钾类玻璃聚类结果.xlsx'，' 铅钡类玻璃聚类分析结果.xlsx'。

表 5　高钾类玻璃聚类结果

文物采样点	01	03 部位 1	03 部位 2	04	05	06 部位 1	06 部位 2	07	09	10	12	13	14	16	18	21	22	27
类别	1	0	1	1	1	1	1	2	2	2	2	1	1	1	0	0	2	2

表 6　铅钡类玻璃聚类结果

文物采样点	02	08	08(2)	11	19	20	23(0)	24	25(0)	26	26(2)	28(0)	29(0)	30_1	30_2	31	32
类别	2	1	1	0	2	0	0	1	0	1	1	3	3	2	2	3	3
文物采样点	33	34	35	36	37	38	39	40	41	42(0)1	42(0)2	43_1	43_2	44(0)	45	46	47
类别	3	2	3	2	3	2	4	4	2	0	0	4	2	3	3	0	0
文物采样点	48	49	49(0)	50	50(0)	51_1	51_2	52	53(0)	54	54(2)	55	56	57	58		
类别	0	2	0	2	0	2	4	2	3	4	4	0	2	2	2		

6.3 合理性与敏感性检验

对于传统的敏感性检验方法，是在原有模型基础上仅对测试样本加入噪声判断其所属分类是否变动，以此考量模型的稳健性，但考虑到传统方法在选择测试样本上的局限性，本文在论文提出的检验模型敏感性方法的基础上[5]，对检验模型敏感性进行了进一步的创新。对所有测试数据加入高斯分布的噪声，重新进行聚类并比较其结果是否发生变化。基于所有测试数据的敏感性检验相比于单独抽取数据的传统敏感性检验，避免了抽样的随机性，获得了更全面的评价。

首先在所有样本数据上均加入不同程度的 Gauss 噪声 $g_k(g_k \sim N(\mu,\sigma_k^2),\sigma_k \in (0.01,5))$，$(k=1,2,\cdots,n)$，得到 n 组样本 $\{X_1,X_2,\cdots,X_n\}$。

具体算法原理如下：

Algorithm 1 生成噪声数据集

Input: X

Output: $X_1,X_2,\cdots,X_n$

Initialization: i←1,$X_i=X,(i\in[1,n])$
for each $\sigma \in (0.01,5)$ **do**
    for each $X_i[i][j]$ in X_i **do**
        $X_i[i][j]$+= random_noise //random_noise $\sim N(0,\sigma^2)$
    end for
end for

其次对得到的 n 组新样本数据进行 KMeans 聚类分析，得到 n 组新的聚类分析结果。不妨记原数据样本为 X，聚类后得到的点集集合记为 C；对于新得到的 n 组样本 $X_1,X_2,\cdots,X_n$，将第 i 组样本聚类分析后得到的点集集合记为 $C_i\,(i=1,2,\cdots,n)$。

若利用传统计算点集之间最近的点的距离衡量两点集之间的差别，则在点集间普遍存在交错时，这一衡量便失去意义；且对于凸包形状不规则的点集，在利用传统质心距

离衡量的过程中也会丧失点集分布的形状等信息，因此我们需要采用一种能保留更多信息度的距离。

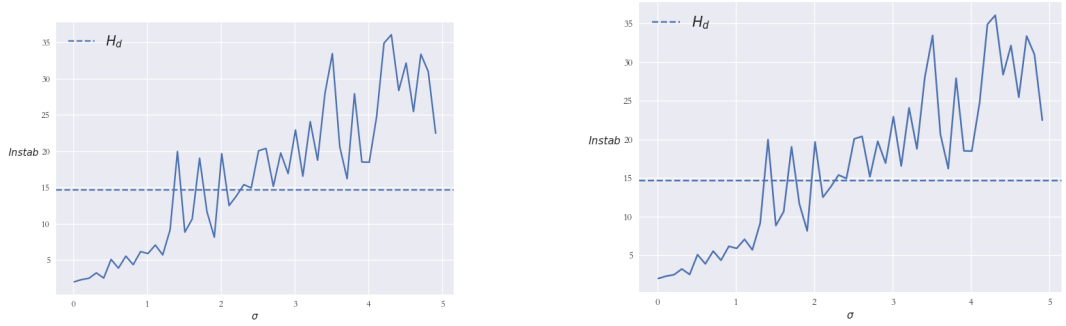
参考文献，我们采用了 Hausdorff 距离作为评价两点集是否接近的指标，这样既能避免点集普遍交错的问题，又能充分考虑点集的形状[6]。

定义

$$\mathrm{Hd}(S, S_i') = \max_{a \in S} \left\{ \min_{b \in S_i'} \{d(a,b)\} \right\}. \tag{3}$$

Hd 刻画了集合 C 中任意点到 S_i' 中最近点距离的最大值，可以作为衡量集合 S,S_i' 距离、偏移量或重合度的重要指标[6]。

不妨设聚类 C 由 t 个集合 S_k 构成，即 $C = \bigcup_{k=1}^{t} S_k$。绘制随着噪声方差 σ 由 0.01 到 5 递增的新聚类 C_i 相对原聚类 C 的累计偏移值 $Instab_{(C,C_i)} = \Sigma_{i\in[1,n]} dis_i$ 曲线，如图8a和8b所示。



(a) 高钾类玻璃稳定性分析　　**(b) 铅钡类玻璃稳定性分析**

图 8　玻璃稳定性分析

在 $\sigma \in [0.01, \sigma_0]$ 范围内，$(0.01 < \sigma_0 \leqslant 5)$, 有

$$\min_{i \neq j, S_i, S_j \in C} \mathrm{Hd}(S_i, S_j) \geqslant Instab_{(C,C_i)}. \tag{4}$$

这就说明加入服从 Gauss 分布方差小于等于 σ_0 的噪声对样本点造成的偏移是有限的，对原有分类方式的改变不大。而在 $\sigma > \sigma_0$ 范围下，噪声的干扰就会对样本点的分类方式造成显著影响。

由图8a和8b可知，在 $\sigma < 1$ 的范围内，噪声的干扰对样本点的分类方式造成的改变不大。而现实中噪声干扰的 σ 通常在 0.05 一下，从而说明模型对于数据的随机误差波动时不敏感的，模型是稳健的。

具体算法原理如下：

Algorithm 2 基于噪声扰动的分类模型敏感性分析

Input: $X, X_1, X_2, ..., X_n$

Output: $Instab_{(C,C_1)}, Instab_{(C,C_2)}, ..., Instab_{(C,C_n)}$

Initialization: 对数据集 $X, X_1, X_2, ..., X_n$，用 Kmeans 进行聚类，得到聚类结果 $C, C_1, C_2, ..., C_n$. C, C_i 都为点集的集合

function Hausdorff_distance(S_1, S_2)

//S_1,S_2 为两个点集,Hausdorff_distance 函数为计算两个点集的距离

  for each $a \in S_1$ **do**

  //对点集 S_1 中的每个点，找到点集 S_2 中欧氏距离距其最近的点 b

    $dis_a \leftarrow \min_{b \in S_2}(\|a - b\|_2)$

  end for //找到两个集合所有相聚最距最近的点的距离最大值

  $dis_{(S1,S2)} \leftarrow \max_{a \in S_1}(dis_a)$

  return $dis_{(S1,S2)}$

end function

for each $C_i, i \in [1, n]$ **do**

//对于每个加入噪声干扰数据 X_i 的聚类结果 C_i, 求其与原数据 X 的聚类结果 C 的差异

  for each $S_k \in C$ **do**

    $dis_{S_k} \leftarrow \min_{S_l \in C_i}$(Hausdorff_distance$(S_k, S_l)$)

  end for

  $Instab_{(C,C_i)} \leftarrow \Sigma_{S_k \in C}(dis_{S_k})$

end for

七、 问题三：未分类样品的鉴别

可根据题意依次对未分类样品进行类型划分与亚类细分，再分析分类结果的敏感性。首先将样品化学成分含量数据代入问题二中建立的 SVM 支撑向量机分类模型，再按玻璃类型代入对应的 KMeans 聚类分析模型，最后在样本数据加入不同程度的 Gauss 噪声，参照原始分类结果得出保持模型分类正确的噪声区间，证明模型稳健性。

7.1 类型与亚类归属

为通过化学成分分析鉴别文物类型，首先考虑将未分类样本点根据玻璃类型分为高钾玻璃和铅钡玻璃两大类。在问题二第一部分中我们建立 SVM 支撑向量机分类模型，求出了可以将不同类型玻璃样本完全分隔的最优支撑超平面，现将未分类样本点各项化学成分数据代入模型，即可通过判断样本点与超平面相对位置得到分类结果。求得的初步分类结果为：

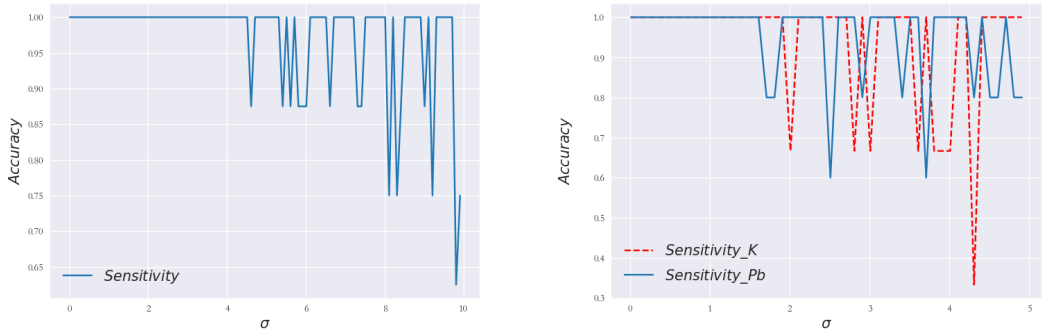
A1、A6、A7 属于高钾玻璃，A2、A3、A4、A5、A8 属于铅钡玻璃。

其次为实现高钾和铅钡样本按亚类的细分，考虑利用问题二第二部分中建立的 KMeans 聚类分析模型，通过将样本点归在距离最近的聚类，实现对样本点所属亚类的判断。求得的细分结果为: A1∈D2，A6、A7∈D0;A4、A8∈C4，A5∈C1

其中 $C_i(i = 1, 2, \cdots, 5)$ 为铅钡玻璃的聚类划分，$D_i(i = 1, 2, 3)$ 为高钾玻璃的聚类划分。

7.2 敏感性分析

首先对判断玻璃类型的 SVM 支撑向量机分类模型进行敏感性分析，结合“采样检测可能导致的偏差通常呈均值为 0 的正态分布”这一生活实际，考虑在未分类样本中加上不同程度的 Gauss 噪声 $t_k(t_k \sim N(0,\sigma_k^2),\sigma_k \in (0.01,10),k=(1,2,\cdots,n)$，将得到的 n 组新数据代入已有的 SVM 支撑向量机分类模型。根据之前的判断结果，定义正确率为“加入噪声后样品点类型判断结果与原样品点类型相同的数目：原未分类样品总数”，绘制随 σ_k 在 (0.01,10) 间增大的判断正确率曲线如图9：



(a) 分类为高钾、铅钡的稳定性　　**(b) 子类稳定性**

图 9　正确率曲线

发现当 $\sigma_k \in (0.01,4.5)$ 时，类型划分模型保持 100% 正确率，说明 SVM 支撑向量机分类模型是稳健的。

其次对判断高钾、铅钡玻璃所属亚类的 KMeans 聚类分析模型进行敏感性分析，出于相同考量，在高钾、铅钡玻璃样品数据上分别加上不同程度的 Gauss 噪声 $r_k(r_k \sim N(0,\sigma_k^2),\sigma_k \in (0.01,5),k=(1,2,\cdots,m)$，将得到的 2m 组新数据代入已有的 KMeans 聚类分析模型。同样绘制出随噪声方差在 (0.01,5) 之间逐步递增的判断正确率曲线如图9a和9b。

发现当 $\sigma_k \in (0.01,1.5)$ 时，高钾玻璃、铅钡玻璃亚类划分模型均保持 100% 正确率，说明的 KMeans 聚类分析模型是稳健的。

八、 问题四：不同类别样品成分关联及差异

为了分析不同类别文物之间化学成分的关联关系，以及关联关系的差异性，考虑到文物的类别如颜色、纹饰可能是多分类定类变量，因此采用方差分析法做出判断，再结合不同种类样品的化学成分含量平均值、标准差等基本信息和 SVM 支持向量机玻璃类型分划模型，综合文献得出关联及差异。

8.1 方差分析模型分类衡量成分差异

可将玻璃文物样品按玻璃类型、颜色、纹饰进行分类，下面分别考量不同种类玻璃文物样品间化学成分差异。不妨以颜色为例，首先根据文物表面的颜色对文物进行分类，由于附件表单 2 中文物一共呈现 8 种颜色，分别是黑色、蓝绿色、绿色、浅蓝色、浅绿色、深蓝色、深绿色、紫色，不妨对这几种颜色编上标号，记黑：1，蓝绿：2，绿：3，浅蓝：4，浅绿：5，深蓝：6，深绿：7，紫：8 以便统计。考虑到附件表单 1 中部分

文物并没有颜色数据，则将这些文物样本数据从附件 2 和附件 1 中剔除，实现数据的预处理。

Step 1.方差分析相关变量

设文物的颜色种类为 D，其中每种颜色划分种类为 $D_1, \cdots, D_8$，对于一种固定的化学成分，令在种类 $D_j(j = 1, 2, \cdots, 8)$ 下的数据（也就是这一种类的文物的这个化学成分含量）的数量为 n_j，再令 E_{ij} 为第 i 个种类的第 i 文物的该化学成分含量，那么得到下表：

表 7　方差分析数据表

	D_1	D_2	$\cdots$	D_8
对应化学成分	E_{11}	E_{12}	$\cdots$	E_{18}
	E_{21}	E_{22}	$\cdots$	E_{28}
	$\vdots$	$\vdots$		$\vdots$
	$E_{n_1 1}$	$E_{n_2 2}$	$\cdots$	$E_{n_8 8}$
样本总和	T_1	T_2	$\cdots$	T_8
样本均值	$\bar{E}_1$	$\bar{E}_2$	$\cdots$	$\bar{E}_8$
总体均值	μ_1	μ_2	$\cdots$	μ_8

其中，令

$$n = n_1 + \cdots + n_8, \tag{5}$$

$$\bar{E}_j = \sum_{l=1}^{8} \sum_{i=1}^{n_j} E_{il} = n\bar{E}。 \tag{6}$$

Step 2. 因素水平

1. 颜色种类为 D_j 的样品化学含量水平 $E_{1j}, \cdots, E_{nj}$ 服从正态分布 $N(\mu_j, \sigma^2)$，容量为 n_j，其中 μ_j, σ 未知。
2. 在不同的样品中抽取的检测样本化学成分含量相互独立。

Step 3. 命题假设检验不同种类玻璃样本化学成分含量均值。

提出原假设为 H_0.

$$H_0 : \mu_1 = \mu_2 = \cdots = \mu_8。 \tag{7}$$

备择假设为 H_1.

$$H_1 : \mu_1, \mu_2, \cdots, \mu_8\text{不全相等}。 \tag{8}$$

Step 4. 模型构建及求解

令

$$S_T = \sum_{k=1}^{8}\sum_{i=1}^{n_j}(E_{ik} - \bar{E})^2, \tag{9}$$

$$S_E = \sum_{k=1}^{8}\sum_{i=1}^{n_j}(E_{ik} - \bar{E}_k)^2, \tag{10}$$

$$S_A = S_T - S_E。 \tag{11}$$

若 H_0 为真，那么检验统计量满足

$$F = \frac{(n-8)S_A}{7S_E} \sim F(s-1, n-s), \tag{12}$$

对于给定的显著性水平 α，得到临界值 F_α，满足

$$P\left\{\frac{(n-8)S_A}{7S_E} > F_\alpha\right\} = \alpha。 \tag{13}$$

8.2 结果分析

通常情况下，取临界值 F_α=0.01，将检验统计量与 F_α 进行比较，得到结果如表8（这里仅截取所有显著性水平小于 0.01 的化学成分，详见支撑材料’ 方差分析纹饰-输出.spv’，’ 方差分析颜色-输出.spv’）：

表 8　颜色分类的方差分析结果

		平方和	自由度	均方	显著性
二氧化硅 (SiO2)	组间	13598.4	6	2266.4	0.000
	组内	23438.3	56	418.541	
氧化铜 (CuO)	组间	137.403	6	22.901	0.000
	组内	192.853	56	3.444	
氧化铅 (PbO)	组间	6628.03	6	1104.67	0.005
	组内	17583.2	56	313.985	
氧化钡 (BaO)	组间	2482.82	6	413.804	0.000
	组内	2435.25	56	43.487	
氧化锶 (SrO)	组间	1.655	6	0.276	0.000
	组内	3.146	56	0.056	
氧化锡 (SnO2)	组间	3.357	6	0.560	0.000
	组内	2.940	56	0.052	
二氧化硫 (SO2)	组间	189.447	6	31.575	0.000
	组内	297.716	56	5.316	

图中表明，SiO_2 等化学元素含量对应的显著性水平小于 0.01，说明拒绝原假设 “不同种类玻璃文物化学元素含量相等”，这表明在不同颜色的玻璃文物中，这些元素的含量差异巨大。

可以推测：这些化学成分含量差异是导致文物呈现不同颜色的原因，结合对数据平均值、标准差、均方等基本数学信息的统计，还可以得出更为具体的信息，如：SrO 组内标准差小，这表明相同颜色玻璃文物的 SrO 含量差异较小。

表 9　纹饰分类的方差分析结果

		平方和	自由度	均方	显著性
氧化钠 (Na2O)	组间	11.5871	2	5.79357	0.13341
	组内	178.34	64	2.78656	
氧化钾 (K2O)	组间	23.9806	2	11.9903	0.47164
	组内	1009.14	64	15.7678	
氧化钙 (CaO)	组间	20.4238	2	10.2119	0.16911
	组内	357.632	64	5.58801	
氧化铁 (Fe2O3)	组间	8.17821	2	4.0891	0.05688
	组内	87.261	64	1.36345	
氧化铜 (CuO)	组间	11.5614	2	5.78071	0.33278
	组内	330.505	64	5.16414	
五氧化二磷 (P2O5)	组间	61.287	2	30.6435	0.10499
	组内	839.853	64	13.1227	
氧化锡 (SnO2)	组间	0.41855	2	0.20928	0.17411
	组内	7.45464	64	0.11648	
二氧化硫 (SO2)	组间	23.4408	2	11.7204	0.20746
	组内	465.295	64	7.27024	

同理，对不同纹饰玻璃样品化学成分含量数据进行方差分析，结果如表9，可以发现符合方差齐性的化学成分所对应的显著性均大于 0.05，说明不同纹饰玻璃文物间化学成分含量无明显差异。

对于不同类型的玻璃样品，在 6.1 中已通过 SVM 分类机模型进行划分，记

$$H(x) = (\boldsymbol{\omega} \cdot \boldsymbol{x}) + b$$

其中 $\boldsymbol{x} \in \mathbb{R}^{14}$,$\boldsymbol{\omega} = (-0.02956241, 0.00551535, -0.05283312, -0.01644926, -0.00040758, 0.03743216, -0.02505811, -0.01378527, 0.10249466, 0.01348478, -0.02241906, 0.00158788, 0, 0)$，

$b = 1.1452272$。

对铅钡玻璃化学成分含量样本点 x_0，有 $H(x_0)$>0;

对高钾玻璃化学成分含量样本点 x_1，有 $H(x_0)$<0。

$\boldsymbol{\omega}$ 中第 3、7、11 项指标均为负，且绝对值较大，说明这些指标对应的化学元素 K_2O、Fe_2O_3、P_2O_5 含量越高 $H(x)$ 越小，样本点越容易出现在超平面下部，玻璃类型越可能为高钾玻璃。

$\boldsymbol{\omega}$ 中第 6、9、10 项指标均为正，且绝对值较大，说明这些指标对应的化学元素 Al_2O_3、PbO、BaO 含量越高 $H(x)$ 越大，样本点越容易出现在超平面上部，玻璃类型越可能为高钾玻璃。

九、 模型的评价与推广

9.1 模型的优缺点

1) 第一问运用变量间相互关系的卡方检验和独立样本的 t 检验，对于关系和统计规律的刻画比较客观和深刻。

2) 第二问使用 SVM 分类机，基于标签数据集进行有监督的学习和分类，大大简化了分类和判断的过程。同时，第二问使用了 KMeans 聚类分析来划分亚类，KMeans 方法找到分割聚类的优化结果，时间复杂度较低，分类算法比较高效，结果比较准确。

3) 第四问用方差分析的方法，检验了以颜色、纹饰为分类依据时处理多个类别间每种化学成分的关联以及差异，具有完整性与直观性。

4) 对风化前玻璃化学成分含量的预测只是得到各个化学成分的范围，未能给出具体的数值，而且 Kmeans 的结果和运行时间与初始质心的选取有一定影响，对于不同的质心选取，结果和效率可能会存在差异。

9.2 模型的推广

第二问中运用 SVM 根据化学成分含量来对古代玻璃类型进行判断并用 KMeans 进行更细致的亚类划分，可以应用于考古学中对出土的未知类型但已知各个化学成分含量的古代玻璃文物进行定量分析的研究，模型对玻璃的类型进行客观的判断和分类，克服了人工分类的不准确性和模糊性。

参考文献

[1] YANG Y. Development history of ancient chinese glass technology ed. by fuxi gan (review) [J]. Technology and Culture, 2022: 597-598.

[2] 毛晓沪. 中国玻璃起源新论[J]. 收藏家, 2016: 58-60.

[3] 刘松, 李青会, 干福熹. 古代玻璃样品表面因素对便携式 X 射线荧光定量分析的影响[J]. 光谱学与光谱分析, 2011, 31(7): 1954-1959.

[4] 孙兆亮司守奎. 数学建模算法与应用（第 2 版）[M]. 国防工业出版社, 2015.

[5] VON LUXBURG U. Clustering stability: An overview[J]. 2010.

[6] HUANG J, CHEN X, JIN H, et al. Automatic classification algorithm for poincare plots based on modified hausdorff distance-support vector machine.[J]. 2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), 2019 12th International Congress on, 2019: 1 - 5.

附录清单

1. 第一问

(1) 铅钡风化.pdf
(2) 铅钡无风化.pdf
(3) 高钾风化.pdf
(4) 高钾无风化.pdf
(5) BD_poly.m
(6) curve_K.m
(7) curve_Pd.m
(8) prediction_K.py
(9) prediction_Pd.py
(10) 高钾无风化.sav
(11) 高钾风化.sav
(12) 铅钡无风化.sav
(13) 铅钡风化.sav
(14) 卡方检验.sav
(15) 高钾 t 检验.sav
(16) 铅钡 t 检验.sav
(17) 高钾无风化-皮尔逊输出.spv
(18) 高钾风化-皮尔逊输出.spv
(19) 铅钡无风化-皮尔逊输出.spv
(20) 铅钡风化-皮尔逊输出.spv
(21) 卡方检验-输出.spv
(22) 高钾 t 检验-输出.spv
(23) 铅钡 t 检验-输出.spv
(24) 皮尔逊高钾无风化输入.xlsx
(25) 皮尔逊高钾风化输入.xlsx
(26) 皮尔逊铅钡无风化输入.xlsx
(27) 皮尔逊铅钡风化输入.xlsx
(28) 频数统计表.xlsx
(29) 历史预测数据输入.xlsx
(30) 高钾类玻璃包络线拟合结果.xlsx
(31) 高钾类玻璃未风化化学成分预测结果.xlsx
(32) 铅钡类玻璃包络线拟合结果.xlsx

(33) 铅钡类玻璃未风化化学成分预测结果.xlsx
(34) 高钾类玻璃包络线拟合.png
(35) 铅钡类玻璃包络线拟合.png

2. 第二问

(1) SVM.pkl
(2) ClusterK.pkl
(3) ClusterPd.pkl
(4) SVM.py
(5) ClusterK.py
(6) ClusterPd.py
(7) ClusterK_steady_analy.py
(8) ClusterPd_steady_analy.py
(9) SVM 输入.xlsx
(10) Cluster 输入.xlsx
(11) 高钾类玻璃聚类结果.xlsx
(12) 铅钡类玻璃聚类结果.xlsx
(13) 高钾类 SSE.png
(14) 高钾类玻璃聚类分析结果.png
(15) 高钾类玻璃稳定性分析.png
(16) 铅钡类 SSE.png
(17) 铅钡类玻璃聚类分析结果.png
(18) 铅钡类玻璃稳定性分析.png

2. 第三问

(1) SVM.pkl
(2) ClusterK.pkl
(3) ClusterPd.pkl
(4) predict_K_Pd.py
(5) predict_K_subgroup.py
(6) predict_Pd_subgroup.py
(7) sensitivity_K_Pd.py
(8) 表单 3.xlsx
(9) 分类为高钾 _ 铅钡的稳定性.png
(10) 子类稳定性.png

3. 第四问

(1) 方差分析纹饰.sav

(2) 方差分析颜色.sav

(3) 方差分析纹饰-输出.spv

(4) 方差分析颜色-输出.spv

附录 A　MATLAB 程序

(1)BD_poly.m

```
% Find the boundary of data
% X-input x coordinary
% Y-input y coordinary
% n-polyfit times
% x_min,x_max,x_int-range of x and interval of x
function [Xup_fit,Yup_fit,Xdown_fit,Ydown_fit,delta_up,delta_down,Aup,Adown]
=BD_poly(X,Y,n_up,n_down,x_min,x_max,x_int,window_n)
[X2,p]=sort(X);
Y2=Y(p);
while mod(length(Y2),window_n)~=0
   Y2(end+1)=0;
   X2(end+1)=0;
end
Y_up=max(reshape(Y2,window_n,length(Y2)/window_n));
for i = 1:length(Y_up)
   X_up(i)=X2(find(Y2==Y_up(i),1));
end
Y_down=min(reshape(Y2,window_n,length(Y2)/window_n));
for i = 1:length(Y_down)
   X_down(i)=X2(find(Y2==Y_down(i),1));
end
[Aup,Sup]=polyfit([X_up],[Y_up],n_up);
Xup_fit=x_min:x_int:x_max;
[Yup_fit,delta_up]=polyval(Aup,Xup_fit,Sup);
[Adown,Sdown]=polyfit([X_down],[Y_down],n_down);
Xdown_fit=x_min:x_int:x_max;
[Ydown_fit,delta_down]=polyval(Adown,Xdown_fit,Sdown);
end
```

(2)curve_K.m

```
data=xlsread('历史预测数据输入.xlsx',4);
%输入所有待拟合的数据
X=data(:,1);
Y_K=data(:,2);
Y_Ca=data(:,3);
Y_Al=data(:,4);
Y_Fe=data(:,5);
%定义拟合的坐标范围和步长
x_min=63;
x_max=95;
x_int=0.01;
%记录所有拟合参数并且写入excel中
curve_fit=[];
%将拟合的参数写入excel
title= ['氧化钾(K2O)上界拟合 ';
     '氧化钾(K2O)下界拟合 ';
```

```
        '氧化钙(CaO)上界拟合 ';
        '氧化钙(CaO)下界拟合 ';
        '氧化铝(Al2O3)上界拟合';
        '氧化铝(Al2O3)下界拟合';
        '氧化铁(Fe2O3)上界拟合';
        '氧化铁(Fe2O3)下界拟合'];
    title=cellstr(title);
    xlswrite('高钾类玻璃包络线拟合结果.xlsx',title,1,'A1')


    %绘制子图1
    subplot(2,2,1)
    n_up=1;
    n_down=1;
    window_n=3;
    [Xup_fit,Yup_fit,Xdown_fit,Ydown_fit,delta_up,delta_down,Aup,Adown]
    =BD_poly(X,Y_K,n_up,n_down,x_min,x_max,x_int,window_n);
    %将拟合的参数写入excel
    xlswrite('高钾类玻璃包络线拟合结果.xlsx',Aup,1,'B1')
    xlswrite('高钾类玻璃包络线拟合结果.xlsx',Adown,1,'B2')
    %开始画图
    scatter(X,Y_K,'xk');
    hold on;
    p1=plot(Xup_fit,Yup_fit,'r--',Xdown_fit,Ydown_fit,'r--');
    p2=plot(Xup_fit,Yup_fit+2*delta_up,'b--');
    %plot(Xdown_fit,Ydown_fit-2*delta_down,'b--');
    legend([p1(1),p2(1)],'包络线拟合','95%置信区间');
    xlabel('二氧化硅(SiO2)含量')
    ylabel('氧化钾(K2O)含量')

    %绘制子图2
    subplot(2,2,2)
    n_up=1;
    n_down=1;
    window_n=4;
    [Xup_fit,Yup_fit,Xdown_fit,Ydown_fit,delta_up,delta_down,Aup,Adown]
    =BD_poly(X,Y_Ca,n_up,n_down,x_min,x_max,x_int,window_n);
    %将拟合的参数写入excel
    xlswrite('高钾类玻璃包络线拟合结果.xlsx',Aup,1,'B3')
    xlswrite('高钾类玻璃包络线拟合结果.xlsx',Adown,1,'B4')
    %开始画图
    scatter(X,Y_Ca,'xk');
    hold on;
    p1=plot(Xup_fit,Yup_fit,'r--',Xdown_fit,Ydown_fit,'r--');
    p2=plot(Xup_fit,Yup_fit+2*delta_up,'b--');
    %plot(Xdown_fit,Ydown_fit-2*delta_down,'b--');
    legend([p1(1),p2(1)],'包络线拟合','95%置信区间');
    xlabel('二氧化硅(SiO2)含量')
    ylabel('氧化钙(CaO)含量')


    %绘制子图3
    subplot(2,2,3)
    n_up=2;
    n_down=2;
    window_n=3;
    [Xup_fit,Yup_fit,Xdown_fit,Ydown_fit,delta_up,delta_down,Aup,Adown]
    =BD_poly(X,Y_Al,n_up,n_down,x_min,x_max,x_int,window_n);
    %将拟合的参数写入excel
    xlswrite('高钾类玻璃包络线拟合结果.xlsx',Aup,1,'B5')
```

```
xlswrite('高钾类玻璃包络线拟合结果.xlsx',Adown,1,'B6')
%开始画图
scatter(X,Y_Al,'xk');
hold on;
p1=plot(Xup_fit,Yup_fit,'r--',Xdown_fit,Ydown_fit,'r--');
p2=plot(Xup_fit,Yup_fit+delta_up,'b--');
plot(Xdown_fit,Ydown_fit-2*delta_down,'b--');
legend([p1(1),p2(1)],'包络线拟合','95%置信区间');
xlabel('二氧化硅(SiO2)含量')
ylabel('氧化铝(Al2O3)含量')


%绘制子图4
subplot(2,2,4)
n_up=2;
n_down=1;
window_n=4;
[Xup_fit,Yup_fit,Xdown_fit,Ydown_fit,delta_up,delta_down,Aup,Adown]
=BD_poly(X,Y_Fe,n_up,n_down,x_min,x_max,x_int,window_n);
%将拟合的参数写入excel
xlswrite('高钾类玻璃包络线拟合结果.xlsx',Aup,1,'B7')
xlswrite('高钾类玻璃包络线拟合结果.xlsx',Adown,1,'B8')
%开始画图
hold on;
scatter(X,Y_Fe,'xk');
p1=plot(Xup_fit,Yup_fit,'r--',Xdown_fit,Ydown_fit,'r--');
p2=plot(Xup_fit,Yup_fit+delta_up,'b--');
plot(Xdown_fit,Ydown_fit-2*delta_down,'b--');
legend([p1(1),p2(1)],'包络线拟合','95%置信区间');
xlabel('二氧化硅(SiO2)含量')
ylabel('氧化铁(Fe2O3)含量')
```

(3)curve_Pd.m

```
data=xlsread('历史预测数据输入.xlsx',3);
%输入所有待拟合的数据
X=data(:,1);
Y_Ca=data(:,2);
Y_Pb=data(:,3);
Y_P=data(:,4);
Y_Sr=data(:,5);
%定义拟合的坐标范围和步长
x_min=0;
x_max=70;
x_int=0.01;
%记录所有拟合参数并且写入excel中
curve_fit=[];
%将拟合的参数写入excel
title= ['氧化钙(CaO)上界拟合 ';
      '氧化钙(CaO)下界拟合 ';
      '氧化铅(PbO)上界拟合 ';
      '氧化铅(PbO)下界拟合 ';
      '五氧化二磷(P2O5)上界拟合';
      '五氧化二磷(P2O5)下界拟合';
      '氧化锶(SrO)上界拟合 ';
      '氧化锶(SrO)下界拟合 '];
title=cellstr(title);
xlswrite('铅钡类玻璃包络线拟合结果.xlsx',title,1,'A1')
```

```
%绘制子图1
subplot(2,2,1)
n_up=2;
n_down=1;
window_n=6;
[Xup_fit,Yup_fit,Xdown_fit,Ydown_fit,delta_up,delta_down,Aup,Adown]
=BD_poly(X,Y_Ca,n_up,n_down,x_min,x_max,x_int,window_n);
%将拟合的参数写入excel
xlswrite('铅钡类玻璃包络线拟合结果.xlsx',Aup,1,'B1')
xlswrite('铅钡类玻璃包络线拟合结果.xlsx',Adown,1,'B2')
%开始画图
scatter(X,Y_Ca,'xk');
hold on;
p1=plot(Xup_fit,Yup_fit,'r--',Xdown_fit,Ydown_fit,'r--');
p2=plot(Xup_fit,Yup_fit+2*delta_up,'b--');
%plot(Xdown_fit,Ydown_fit-2*delta_down,'b--');
legend([p1(1),p2(1)],'包络线拟合','95%置信区间');
xlabel('二氧化硅(SiO2)含量')
ylabel('氧化钙(CaO)含量')

%绘制子图2
subplot(2,2,2)
n_up=2;
n_down=2;
window_n=3;
[Xup_fit,Yup_fit,Xdown_fit,Ydown_fit,delta_up,delta_down,Aup,Adown]
=BD_poly(X,Y_Pb,n_up,n_down,x_min,x_max,x_int,window_n);
%将拟合的参数写入excel
xlswrite('铅钡类玻璃包络线拟合结果.xlsx',Aup,1,'B3')
xlswrite('铅钡类玻璃包络线拟合结果.xlsx',Adown,1,'B4')
%开始画图
scatter(X,Y_Pb,'xk');
hold on;
p1=plot(Xup_fit,Yup_fit,'r--',Xdown_fit,Ydown_fit,'r--');
p2=plot(Xup_fit,Yup_fit+2*delta_up,'b--');
%plot(Xdown_fit,Ydown_fit-2*delta_down,'b--');
legend([p1(1),p2(1)],'包络线拟合','95%置信区间');
xlabel('二氧化硅(SiO2)含量')
ylabel('氧化铅(PbO)含量')


%绘制子图3
subplot(2,2,3)
n_up=2;
n_down=1;
window_n=6;
[Xup_fit,Yup_fit,Xdown_fit,Ydown_fit,delta_up,delta_down,Aup,Adown]
=BD_poly(X,Y_P,n_up,n_down,x_min,x_max,x_int,window_n);
%将拟合的参数写入excel
xlswrite('铅钡类玻璃包络线拟合结果.xlsx',Aup,1,'B5')
xlswrite('铅钡类玻璃包络线拟合结果.xlsx',Adown,1,'B6')
%开始画图
scatter(X,Y_P,'xk');
hold on;
p1=plot(Xup_fit,Yup_fit,'r--',Xdown_fit,Ydown_fit,'r--');
p2=plot(Xup_fit,Yup_fit+2*delta_up,'b--');
%plot(Xdown_fit,Ydown_fit-2*delta_down,'b--');
legend([p1(1),p2(1)],'包络线拟合','95%置信区间');
xlabel('二氧化硅(SiO2)含量')
ylabel('五氧化二磷(P2O5)含量')
```

```
%绘制子图4
subplot(2,2,4)
n_up=1;
n_down=1;
window_n=7;
[Xup_fit,Yup_fit,Xdown_fit,Ydown_fit,delta_up,delta_down,Aup,Adown]
=BD_poly(X,Y_Sr,n_up,n_down,x_min,x_max,x_int,window_n);
%将拟合的参数写入excel
xlswrite('铅钡类玻璃包络线拟合结果.xlsx',Aup,1,'B7')
xlswrite('铅钡类玻璃包络线拟合结果.xlsx',Adown,1,'B8')
%开始画图
hold on;
scatter(X,Y_Sr,'xk');
p1=plot(Xup_fit,Yup_fit,'r--',Xdown_fit,Ydown_fit,'r--');
p2=plot(Xup_fit,Yup_fit+2*delta_up,'b--');
%plot(Xdown_fit,Ydown_fit-2*delta_down,'b--');
legend([p1(1),p2(1)],'包络线拟合','95%置信区间');
xlabel('二氧化硅(SiO2)含量')
ylabel('氧化锶(SrO)含量')
```

附录 B　Python 代码

(1)prediction_K.py

```
from calendar import c
import numpy as np
import pandas as pd
import math
import os

data = pd.read_excel('历史预测数据输入.xlsx', sheet_name='高钾风化待预测')
p = pd.read_excel('高钾类玻璃包络线拟合结果.xlsx', header=None) # p为待拟合的参数
Si_range = [60, 77]

column = ['氧化钾(K2O)',
       '氧化钙(CaO)',
       '氧化铝(Al2O3)',
       '氧化铁(Fe2O3)']
predict_data = pd.DataFrame()

average = [0.800942, 0.277524, 0.028252]
for i, col in enumerate(column):
   n_up = p.iloc[2*i].count()-2 # 该行非空值的个数，最后的常数项重新计算
   n_down = p.iloc[2*i+1].count()-2

   # 计算上界的系数
   b_up = data[col].copy()
   for j in range(1, n_up+1):
      b_up -= p[j][2*i]*np.power(data['二氧化硅(SiO2)'], n_up-j+1)
   # 计算上界的上范围
   b_up1 = b_up.copy()
   b_up2 = b_up.copy()
   for j in range(1, n_up+1):
      b_up1 += p[j][2*i]*math.pow(Si_range[0], n_up-j+1)
   for j in range(1, n_up+1):
      b_up2 += p[j][2*i]*math.pow(Si_range[1], n_up-j+1)
```

```
        b_up = b_up2.combine(b_up1, np.maximum).copy()

        # 计算下界的系数
        b_down = data[col].copy()
        for j in range(1, n_down+1):
            b_down -= p[j][2*i+1]*np.power(data['二氧化硅(SiO2)'], n_down-j+1)
        # 计算下界的上范围
        b_down1 = b_down.copy()
        b_down2 = b_down.copy()
        for j in range(1, n_down+1):
            b_down1 += p[j][2*i+1]*math.pow(Si_range[0], n_down-j+1)
        for j in range(1, n_down+1):
            b_down2 += p[j][2*i+1]*math.pow(Si_range[1], n_down-j+1)
        b_down = b_down2.combine(b_down1, np.minimum).copy()

        predict_data[col+'上界'] = b_up.copy()
        predict_data[col+'下界'] = b_down.copy()

    # 保持不变的项以及文物的标签
    col_static = ['文物采样点',
                  '氧化钠(Na2O)',
                  '氧化铜(CuO)',
                  '氧化钡(BaO)',
                  '五氧化二磷(P2O5)',
                  '氧化锡(SnO2)'
                 ]
    for col in col_static:
        predict_data[col] = data[col]
    # 求平均的项
    predict_data['氧化镁(MgO)'] = average[0]
    predict_data['氧化铅(PbO)'] = average[1]
    predict_data['氧化锶(SrO)'] = average[2]
    # 二氧化硫为0
    predict_data['二氧化硫(SO2)'] = 0
    # 二氧化硅的范围
    predict_data['二氧化硅(SiO2)上界'] = Si_range[1]
    predict_data['二氧化硅(SiO2)下界'] = Si_range[0]


    # 对数据按照原来的顺序重新排序
    predict_data = predict_data[['文物采样点', '二氧化硅(SiO2)上界', '二氧化硅(SiO2)下界',
        '氧化钠(Na2O)', '氧化钾(K2O)上界', '氧化钾(K2O)下界',
                         '氧化钙(CaO)上界', '氧化钙(CaO)下界', '氧化镁(MgO)',
                        '氧化铝(Al2O3)上界', '氧化铝(Al2O3)下界', '氧化铁(Fe2O3)上界',
                            '氧化铁(Fe2O3)下界',
                         '氧化铜(CuO)', '氧化铅(PbO)', '氧化钡(BaO)', '五氧化二磷(P2O5)',
                         '氧化锶(SrO)', '氧化锡(SnO2)', '二氧化硫(SO2)']
                        ]

    # 写入文件
    pd.DataFrame(predict_data.to_excel('高钾类玻璃未风化化学成分预测结果.xlsx'))
```

(2)prediction_Pd.py

```
    from calendar import c
    import numpy as np
    import pandas as pd
    import math
    import os
```

```
data = pd.read_excel('历史预测数据输入.xlsx', sheet_name='铅钡风化待预测')
p = pd.read_excel('铅钡类玻璃包络线拟合结果.xlsx', header=None) # p为待拟合的参数
Si_range = [50,70]

column = ['氧化钙(CaO)',
       '氧化铅(PbO)',
       '五氧化二磷(P2O5)',
       '氧化锶(SrO)']
predict_data = pd.DataFrame()

average = [0.92721171]
for i, col in enumerate(column):
   n_up = p.iloc[2*i].count()-2 # 该行非空值的个数，最后的常数项重新计算
   n_down = p.iloc[2*i+1].count()-2

   # 计算上界的系数
   b_up = data[col].copy()
   for j in range(1, n_up+1):
      b_up -= p[j][2*i]*np.power(data['二氧化硅(SiO2)'], n_up-j+1)
   # 计算上界的上范围
   b_up1 = b_up.copy()
   b_up2 = b_up.copy()
   for j in range(1, n_up+1):
      b_up1 += p[j][2*i]*math.pow(Si_range[0], n_up-j+1)
   for j in range(1, n_up+1):
      b_up2 += p[j][2*i]*math.pow(Si_range[1], n_up-j+1)
   b_up = b_up2.combine(b_up1, np.maximum).copy()

   # 计算下界的系数
   b_down = data[col].copy()
   for j in range(1, n_down+1):
      b_down -= p[j][2*i+1]*np.power(data['二氧化硅(SiO2)'], n_down-j+1)
   # 计算下界的上范围
   b_down1 = b_down.copy()
   b_down2 = b_down.copy()
   for j in range(1, n_down+1):
      b_down1 += p[j][2*i+1]*math.pow(Si_range[0], n_down-j+1)
   for j in range(1, n_down+1):
      b_down2 += p[j][2*i+1]*math.pow(Si_range[1], n_down-j+1)
   b_down = b_down2.combine(b_down1, np.minimum).copy()

   predict_data[col+'上界'] = b_up.copy()
   predict_data[col+'下界'] = b_down.copy()

# 保持不变的项以及文物的标签
col_static = ['文物采样点',
         '氧化钾(K2O)',
         '氧化镁(MgO)',
         '氧化铝(Al2O3)',
         '氧化铁(Fe2O3)',
         '氧化铜(CuO)',
         '氧化钡(BaO)',
         '氧化锡(SnO2)'
         ]
for col in col_static:
   predict_data[col] = data[col]
# 求平均的项
predict_data['氧化钠(Na2O)'] = average[0]
# 二氧化硫为0
predict_data['二氧化硫(SO2)'] = 0
```

```
# 二氧化硅的范围
predict_data['二氧化硅(SiO2)上界'] = Si_range[1]
predict_data['二氧化硅(SiO2)下界'] = Si_range[0]


# 对数据按照原来的顺序重新排序
predict_data = predict_data[['文物采样点', '二氧化硅(SiO2)上界', '二氧化硅(SiO2)下界',
    '氧化钠(Na2O)', '氧化钾(K2O)',
                '氧化钙(CaO)上界', '氧化钙(CaO)下界', '氧化镁(MgO)', '氧化铝(Al2O3)',
                    '氧化铁(Fe2O3)',
                '氧化铜(CuO)', '氧化铅(PbO)上界', '氧化铅(PbO)下界',
                    '氧化钡(BaO)','五氧化二磷(P2O5)上界',
                '五氧化二磷(P2O5)下界', '氧化锶(SrO)上界', '氧化锶(SrO)下界',
                    '氧化锡(SnO2)', '二氧化硫(SO2)']
               ]

# 写入文件
pd.DataFrame(predict_data.to_excel('铅钡类玻璃未风化化学成分预测结果.xlsx'))
```

(3) ClusterK.py

```
import pandas as pd
import numpy as np
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt
import os
import joblib

plt.style.use('seaborn')
# 显示汉字 SimHei黑体, STsong 华文宋体还有font.style font.size等
plt.rcParams['font.family'] = 'STsong'
plt.rcParams['axes.unicode_minus'] = False

# 定义数据集
data=pd.read_excel("Cluster输入.xlsx",sheet_name="高钾")
col = data.columns.values.tolist()
col.remove('表面风化')
col.remove('采样类型')
col.remove('文物采样点')
#获得待聚类的数据
X = data[col].copy()
#plt.scatter(X[col[0]],X[col[1]])
#plt.show()

sse = []
for k in range(1,9):
    estimator = KMeans(n_clusters=k)
    estimator.fit(X)
    sse.append(estimator.inertia_)
fig,ax = plt.subplots()
ax.plot(np.arange(1,9),sse,marker='o')
plt.xlabel('分类数目')
plt.ylabel('SSE')
plt.show()

model = KMeans(n_clusters=3)
model.fit(X)
# 为每个示例分配一个集群
yhat = model.predict(X)
plt.scatter(X[col[0]],X[col[1]],c=yhat,cmap='Dark2')
```

```
    plt.xlabel('二氧化硅(SiO2)')
    plt.ylabel('氧化钾(K2O)')
    label = data['文物采样点'].tolist()
    for i in range(len(label)):
        plt.text(X[col[0]][i], X[col[1]][i], label[i],fontsize = 10)
    plt.show()

    data.insert(1,'类别',yhat)
    #写入文件i
    pd.DataFrame(data.to_excel("高钾类玻璃聚类结果.xlsx"))

    # 保存训练好的模型
    joblib.dump(model,'ClusterK.pkl')
```

(4)ClusterK_steady_analy.py

```
import pandas as pd
import numpy as np
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt
import random
from scipy.spatial.distance import cdist

plt.style.use('seaborn')
# 显示汉字 SimHei黑体, STsong 华文宋体还有font.style font.size等
plt.rcParams['font.family'] = 'STsong'
plt.rcParams['axes.unicode\_minus'] = False


def Hausdorff\_distance(X1, X2):
    # 计算集合每两点之间的欧式距离
    dis = cdist(X1, X2, metric='euclidean')
    # 求出每一列的最小值, 即min{d(a,b)}
    temp1 = np.min(dis, axis=1)
    # 求出每一行的最大值, 即max{min{d(a,b)}}
    temp2 = np.max(temp1, axis=0)
    return temp2


# 定义数据集
data = pd.read\_excel("Cluster输入.xlsx", sheet\_name="铅钡")
col = data.columns.values.tolist()
col.remove('表面风化')
col.remove('采样类型')
col.remove('文物采样点')
# 获得待聚类的数据
X = data[col].copy()

# 对数据进行直接聚类的结果
model\_origin = KMeans(n\_clusters=5)
model\_origin.fit(X)
# 为每个示例分配一个集群
yhat\_origin = model\_origin.predict(X)
X['cluster'] = yhat\_origin

instability = []
for sigma in np.arange(0.01, 5, 0.1):
    # 给数据添加高斯噪声, 以检验分类模型的灵敏度
    X\_noise = X.copy()
    # 未每一个数据添加高斯噪声
```

```
    for u in col:
        for i in range(X\_noise.shape[0]):
            mu = 0
            X\_noise[u][i] += random.gauss(mu, sigma)

    model\_noise = KMeans(n\_clusters=5)
    model\_noise.fit(X\_noise)
    # 为每个示例分配一个集群
    yhat\_noise = model\_noise.predict(X\_noise)
    X\_noise['cluster'] = yhat\_noise

    # 求出最小的Hausdorff距离，即求出了分类最为接近的点集，
    将所有集合间两两最小的Hausdorff距离相加，即可以评价两个聚类模型的相似性
    dis = []
    for i in range(5):
        select\_rows\_origin = [c for c in X.index if X['cluster'][c] == i]
        select\_cluster\_origin = np.array(X.loc[select\_rows\_origin, :])
        line = []
        for j in range(5):
            select\_rows\_noise = [
                c for c in X\_noise.index if X\_noise['cluster'][c] == j]
            select\_cluster\_noise = np.array(X\_noise.loc[select\_rows\_noise, :])
            d = Hausdorff\_distance(select\_cluster\_noise, select\_cluster\_origin)
            line.append(d)
        dis.append(line)

    # 找到距离最近的集合
    temp = np.min(dis, axis=1)
    # 将最近的集合距离进行相加
    instability.append(np.sum(temp))

# 求出原集合cluster彼此的最近距离
dis\_ = []
for i in range(5):
    select\_rows = [c for c in X.index if X['cluster'][c] == i]
    select\_cluster = np.array(X.loc[select\_rows, :])
    for j in range(i+1, 5):
        select\_rows\_ = [c for c in X.index if X['cluster'][c] == j]
        select\_cluster\_ = np.array(X.loc[select\_rows\_, :])
        d = Hausdorff\_distance(select\_cluster\_, select\_cluster)
        dis\_.append(d)

min\_dis\_ = np.min(dis\_)

# 做出sigma和Instab的关系，并且最后将分界线min\_dis\_画在图上
plt.plot(np.arange(0.01, 5, 0.1), instability)
line1 = plt.axhline(min\_dis\_, linestyle='--')
plt.legend(handles=[line1], labels=[r'$H\_d$'], loc="upper left",
        fontsize=15)
plt.xlabel(r'$\sigma$')
plt.ylabel(r'$Instab$', rotation=0, labelpad=20)
plt.show()

print(min\_dis\_)
```

(5)SVM.py

```
import pandas as pd
from sklearn.svm import SVC
from matplotlib import pyplot as plt
```

```
import numpy as np
from sklearn.model\_selection import train\_test\_split
import joblib


# 从文件中读取数据
data = pd.read\_excel("SVM输入.xlsx")
# 将读取的data划分为x,y;
col = data.columns.values.tolist()
col.remove('类型')
X = np.array(data[col].copy())
y = np.array(data['类型'])
#将数据随机的划分为测试集和训练集
train\_X,test\_X, train\_y, test\_y = train\_test\_split(X,y,test\_size =
    0.2,random\_state = 0)
# 进行分类器的训练
svc = SVC(kernel='linear')
svc.fit(train\_X, train\_y)
print('正确率')
print(svc.score(test\_X,test\_y))
pred\_y=svc.predict(test\_X)

# from sklearn.metrics import roc\_curve
# import matplotlib.pyplot as plt

# ## 求出ROC曲线的x轴和y轴
# fpr, tpr, thresholds = roc\_curve(test\_y,pred\_y)
# #绘制ROC曲线
# plt.figure(figsize=(10,6))
# plt.xlim(0,1)  ##设定x轴的范围
# plt.ylim(0.0,1.1) ## 设定y轴的范围
# plt.xlabel('False Postive Rate')
# plt.ylabel('True Postive Rate')
# plt.plot(fpr,tpr,linewidth=2, linestyle="-",color='red')
# plt.show()

#获得线性分类平面的方程,WX+b=0
w = svc.coef\_
b = svc.intercept\_
print("超平面方程")
print(w)
print(b)

# 保存训练好的模型
joblib.dump(svc,'SVM.pkl')
```

(6)predict_K_Pd.py

```
import joblib
import pandas as pd
import numpy as np

# plt.style.use('seaborn')
# # 显示汉字 SimHei黑体, STsong 华文宋体还有font.style font.size等
# plt.rcParams['font.family'] = 'STsong'
# plt.rcParams['axes.unicode\_minus'] = False

# 调用保存好的模型(训练好的)去做预测
model = joblib.load("SVM.pkl")
data = pd.read\_excel('表单3.xlsx',sheet\_name='归一化的表单3')
```

```
#获得待分析的数据
col = data.columns.values.tolist()
col.remove('文物编号')
col.remove('表面风化')
X = data[col].copy()

y\_predict = model.predict(np.array(X))
print(y\_predict)
```

(7)predict_K_subgroup.py

```
import joblib
import pandas as pd

# 调用保存好的模型(训练好的)去做预w
model = joblib.load("ClusterK.pkl")
data = pd.read\_excel('表单3.xlsx',sheet\_name='高钾类玻璃')
#获得待分析的数据
col = data.columns.values.tolist()
col.remove('文物编号')
col.remove('表面风化')
X = data[col].copy()

y\_predict = model.predict(X)
print(y\_predict)
```

(8)predict_Pd_subgroup.py

```
import joblib
import pandas as pd

# plt.style.use('seaborn')
# # 显示汉字 SimHei黑体, STsong 华文宋体还有font.style font.size等
# plt.rcParams['font.family'] = 'STsong'
# plt.rcParams['axes.unicode\_minus'] = False

# 调用保存好的模型(训练好的)去做预测
model = joblib.load("ClusterPd.pkl")
data = pd.read\_excel('表单3.xlsx',sheet\_name='铅钡类玻璃')
#获得待分析的数据
col = data.columns.values.tolist()
col.remove('文物编号')
col.remove('表面风化')
X = data[col].copy()

y\_predict = model.predict(X)
print(y\_predict)
```

(8)sensitivity_K_Pd.py

```
import joblib
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import random
from sklearn.metrics import accuracy\_score

plt.style.use('seaborn')
# 显示汉字 SimHei黑体, STsong 华文宋体还有font.style font.size等
```

```
plt.rcParams['font.family'] = 'STsong'
plt.rcParams['axes.unicode\_minus'] = False
random.seed(61)

# 调用保存好的模型(训练好的)去做预测
model\_K\_Pd = joblib.load("SVM.pkl")
model\_K = joblib.load('ClusterK.pkl')
model\_Pd = joblib.load('ClusterPd.pkl')
data1 = pd.read\_excel('表单3.xlsx', sheet\_name='归一化的表单3')
data2 = pd.read\_excel('表单3.xlsx', sheet\_name='高钾类玻璃')
data3 = pd.read\_excel('表单3.xlsx', sheet\_name='铅钡类玻璃')
# 删除无关数据的列
col1 = data1.columns.values.tolist()
col1.remove('文物编号')
col1.remove('表面风化')
col2 = ['二氧化硅(SiO2)', '氧化钾(K2O)']
col3 = ['二氧化硅(SiO2)', '氧化铅(PbO)', '氧化钡(BaO)']
# 拷贝待分析的数据
X1 = data1[col1].copy() # X1为分为高钾和铅钡类玻璃所需要的数据
X2 = data2[col2].copy() # X2为高钾玻璃分亚类所需要的数据
X3 = data3[col3].copy() # X3为铅钡玻璃分亚类所需要的数据

y1\_pred = model\_K\_Pd.predict(np.array(X1))
y2\_pred = model\_K.predict(X2)
y3\_pred = model\_Pd.predict(X3)


sensitivity1 = []
sensitivity2 = []
sensitivity3 = []


for sigma in np.arange(0.01, 10, 0.1):
    # 给数据添加高斯噪声，以检验分类模型的灵敏度
    X1\_noise = X1.copy()
    # 为每一个数据添加高斯噪声
    for u in col1:
        for i in range(X1\_noise.shape[0]):
            X1\_noise[u][i] += random.gauss(0, sigma)
    # 为每个示例分配一个集群
    y1\_pred\_noise = model\_K\_Pd.predict(np.array(X1\_noise))
    # 计算加入噪声后的预测准确率
    acc1 = accuracy\_score(y1\_pred, y1\_pred\_noise)
    sensitivity1.append(acc1)


for sigma in np.arange(0.01, 5, 0.1):
    # 给数据添加高斯噪声，以检验分类模型的灵敏度
    X2\_noise = X2.copy()
    X3\_noise = X3.copy()
    # 为每一个数据添加高斯噪声
    for u in col2:
        for i in range(X2\_noise.shape[0]):
            X2\_noise[u][i] += random.gauss(0, sigma)
    for u in col3:
        for i in range(X3\_noise.shape[0]):
            X3\_noise[u][i] += random.gauss(0, sigma)

    # 为每个示例分配一个集群
    y2\_pred\_noise = model\_K.predict(X2\_noise)
```

```
        y3\_pred\_noise = model\_Pd.predict(X3\_noise)

        # 计算加入噪声后的预测准确率
        acc2 = accuracy\_score(y2\_pred, y2\_pred\_noise)
        acc3 = accuracy\_score(y3\_pred, y3\_pred\_noise)
        sensitivity2.append(acc2)
        sensitivity3.append(acc3)

    # 开始绘图
    plt.figure()
    x = np.arange(0.01, 10, 0.1)
    ax1, = plt.plot(x, sensitivity1,'#1f77b4')
    plt.legend(handles=[ax1], labels=[r'$Sensitivity$'], loc="best",
            fontsize=15)
    plt.xlabel(r'$\sigma$', fontsize=15)
    plt.ylabel(r'$Accuracy$', fontsize=15)
    # 亚类画做一张图
    plt.figure()
    x1 = np.arange(0.01, 5, 0.1)
    ax2, = plt.plot(x1, sensitivity2,'r',linestyle='--')
    ax3, = plt.plot(x1, sensitivity3,'#1f77b4')
    plt.legend(handles=[ax2, ax3], labels=[r'$Sensitivity\\_K$', r'$Sensitivity\\_Pb$'],
        loc="best",
            fontsize=15)
    plt.xlabel(r'$\sigma$', fontsize=15)
    plt.ylabel(r'$Accuracy$', labelpad=20, fontsize=15)
    plt.show()
```

附录 C　Excel 图表

(1) 高钾类玻璃包络线拟合结果

氧化钾(K2O)上界拟合	-0.39275	38.71277	
氧化钾(K3O)上界拟合	-0.3108	27.50501	
氧化钾(K4O)上界拟合	-0.2407	23.73744	
氧化钾(K5O)上界拟合	-0.06839	6.168547	
氧化钾(K6O)上界拟合	-0.00526	0.569273	-4.87978
氧化钾(K7O)上界拟合	0.000145	-0.16365	15.56909
氧化钾(K8O)上界拟合	0.002767	-0.56474	29.02766
氧化钾(K9O)上界拟合	-0.01492	1.371172	

(2) 铅钡类玻璃包络线拟合结果

氧化钙(CaO)上界拟合	-0.00061	-0.02916	6.493919
氧化钙(CaO)下界拟合	0.004909	0.25342	
氧化铅(PbO)上界拟合	0.003257	-1.13898	81.04876
氧化铅(PbO)下界拟合	-0.01846	1.079242	18.04178
五氧化二磷(P2O5)上界拟合	0.002049	-0.45591	22.62081
五氧化二磷(P2O5)下界拟合	0.016274	0.081324	
氧化锶(SrO)上界拟合	-0.01349	1.273419	
氧化锶(SrO)下界拟合	0.002906	0.037155	