队伍编号	21230010021
题号	B

基于随机森林与团簇几何构型的最优结构预测探索

摘 要

为了解决应用传统理论计算团簇全局最优结构时迭代次数过多，且高精度的理论计算时间呈指数增长的问题，我们欲引入机器学习算法辅助求解团簇全局最优结构，从而更有助于发现新型团簇材料的结构和性能。

针对问题一给出的 1000 个金团簇Au_{20}的结构，我们进行数据剥离，循环读取文件生成新的数据表单，从而直观的看出了团簇能量与团簇内各原子坐标之间的关系。同时，我们引入辅助量**原子密度、团簇体积**来**特征化**地表示其对于能量值的影响，并带入 **LJ 团簇势能公式**，发现其波动与能量曲线波动一致，证实了 AU 团簇的能量构成主要来源为**势能**。针对预测模型，我们重点比对了 **BP 神经网络**与**随机森林算法**的优劣，最终选取随机森林算法构建 62 个特征值的**预测模型**。为了缩小搜索量，我们采用 **K-means 算法**进行聚类，并以 **SVM 分类器**辅助验证聚类效果良好。从而将坐标**搜索跨度**减小为特定类别的各项坐标跨度，很大程度减轻了搜索原理。最终随机生成 **100 万**条数据成功构建出了较为优化的Au_{20}构型：**类正四面体构型**。

针对问题二求解Au_{32}的全局最优结构，由于第一问得到的模型为输入端 62 个数据，因而无法直接带入求解。我们欲转化Au_{32}的 96 个坐标，进行**降维**，循环转化为 60 个坐标，分批次带入一种模型得到预测能量值，取预测值的均值为最终预测结果，从而构建出了一组较优坐标值。我们推测Au_{32}的全局最优结构为类似足球的**多个六边形组成的正多面体**。考虑到问题一的最优构型为正四面体，我们在此基础上**对四面体进行扩充**，不断构造新的四面体并得到多组**异构体**从中筛选出最优。

针对问题三、四，我们仿照一、二的求解流程，考虑了硼团簇的**电负性**对于能量值造成的影响以及关于B_{40}^-的处理方式，考虑对预测模型**引入 5 组无关坐标**，保持输入端一致进行预测，我们推测硼团簇B_{45}^-的全局最优结构为**多折叠、无空缺类型**；B40-的全局最优结构为**对称**且具备两个六边形的孔洞。

关键词：K-means 聚类 随机森林 搜索跨度 BP SVM 几何构型

目录

一、问题背景

团簇是由几个至上千个原子、分子或离子组成的相对稳定的微观或亚微观聚集体，是介于原子、分子与宏观的固体物质之间的一种新层次，具有许多奇特的性质。团簇的结构和性质与其包含原子的数目和种类密切，其物理和化学性质随所包含的原子数目而变化。其中，金属团簇由于其独特的催化性质、光学性质和结构，受到研究者的广泛关注。硼团簇的几何结构往往较为复杂，单纯的实验很难直接确定硼团簇的精确结构。因此，硼团簇的研究往往需要理论与实验想结合的方法。为了更好地了解团簇的性质及结构、预测新型团簇的全局最优结构，我们需要深入研究团簇的结构优化问题——即是团簇最低能量结构的预测问题。然而，由于其构型空间随团簇尺度的增加而呈指数型增长。故对于此类问题，发展高效的全局优化算法具有重要的理论和现实意义。

二、问题分析

2.1 问题一的分析

针对问题一，首先，我们将.xyz 文件转换为.txt 文件，把 999 个数据文件输入 python 获取训练集（999×62）。接下来，我们采用机器学习算法，对比了 logistics、LSTM、BP 神经网络、随机森林等算法，最后选用随机森林算法测算。我们采用随机模拟的方法生成多条数据，预测全局最优结构。具体后文有阐释。

2.2 问题二的分析

考虑到问题一的输入集度量为 60 个坐标，而我们需要 96 个坐标。因此，我们考虑把 96 个坐标转换为 60 个坐标，分批次输入训练，选取每组预测值均值为预测结果，从而得到最优坐标解，借用 Matlab 画出三维图。

2.3 问题三的分析

仿照问题一，我们发现数据集扩倍较大，并引入了电负性的概念进一步优化求解。

2.4 问题四的分析

仿照问题二，欲设置 5 个无关坐标，仍保持输入集为 45 进行预测。

三、基本假设

为了使得问题更易于理解，我们作出以下合理假设：

1. 假设团簇的能量仅受原子密度、原子体积和附件中所给各原子坐标影响。
2. 假设所选指标能够代表所要研究对象的整体情况。
3. 假设各坐标对欧氏距离的贡献是同等的，欧氏距离效果理想。
4. 假设量化处理的数据无错误无丢失。
5. 忽略相对论效应的影响。
6. 假设题目给出的数据真实可靠。

四、问题一的模型建立与求解

针对问题一，关于预测团簇分子、结构的问题，我们采用多种预测模型进行比较分析。考虑到现有结合机器学习的预测团簇构型方法较少，我们查阅大量文献，结合金属团簇的物理、化学性质，选取了影响Au团簇能量的相关指标，剔除粘合性，尽量保持变量的独立性，构建了较为完善的预测模型，并预测其全局最优结构，同时以三维结构图呈现。

4.1 数据预处理

通过分析比对，发现附件只给出 999 个金团簇数据文件，第 155 号数据缺失。我们分别对坐标数据、能量数据、原子名称进行分离，得到了999×62的数据表单。对能量值进行画图，得到能量值域。取前十与后十个数据，载入 VMD 进行直观分析，对金团簇的构型有初步认知。

4.2 模型的建立

我们查阅文献，了解到影响团簇稳定性的因素——平均结合能，能隙值、电负性（排斥力、吸引力）、能量二阶差分、平均键长等。考虑到已知数据较少，我们引入原子密度、原子体积进行辅助分析。

基于论文，对于许多大尺度 LJ (Lennard-Jones)实例，其最低能量构型是无中心原子的二十面体。我们想要求得最稳定、能量最低的团簇构型，那么可知团簇的中心原子越少越好。处于内层结构的原子受到外层原子的挤压，内层原子间的距离比外层原子间的距离更小，因此，越内层的原子，原子密度越大。我们可以得出，团簇内两两原子间距越小，其密度越大，金属键结合能力更强，团簇的热力学稳定性越高。

我们将原子i的密度定义为

$$D(i) = \sum_{\substack{j=1(\neq i) \\ d_{ij} \leq \sqrt[6]{2}}}^{N} \frac{1}{d_{ij}^3} \tag{1}$$

其中，d_{ij}表示原子i与原子j之间的距离，N为团簇尺度，此处统一为20。

对于体积的计算，我们在三维空间内可以将含n个原子的团簇放入长、宽、高分别为x_1，y_1，z_1的长方体（或立方体）空间，该团簇体积应小于此长方体体积。设团簇的体积为V_1，长方体的体积为$V_2 = x_1 \times y_1 \times z_1$。用统计方法随机地在$V_2$范围内产生$n$个三维空间内的点，$n$有$50\%$的概率落在$V_1$内或$V_1$外。我们把产生的$m$个点看作随机变量，则$n$个点在$V_2$范围内为均匀分布。设落在$V_1$内的点数为$n_{V_1}$。综上，我们用随机模拟的统计方法求解，得到体积计算式为

$$V_1 = \left(\frac{n_{V_1}}{n}\right) V_2 \tag{2}$$

我们假定，同类型Au_{20}之间的原子运动速率基本一致。使得随机模拟的概率保持一致。

我们令x_1，y_1，z_1为团簇内原子在x、y、z轴上的最大跨度，得到计算公式（ε为干扰项）如下：

$$x_1 = \max\{\mathrm{atom}_x(x,y,z)\} - \min\{\mathrm{atom}_x(x,y,z)\} + \varepsilon \tag{3}$$

$$y_1 = \max\{\mathrm{atom}_y(x,y,z)\} - \min\{\mathrm{atom}_y(x,y,z)\} + \varepsilon \tag{4}$$

$$z_1 = \max\{\mathrm{atom}_z(x,y,z)\} - \min\{\mathrm{atom}_z(x,y,z)\} + \varepsilon \tag{5}$$

查阅文献，我们引入 LJ 团簇的势能函数。表示如下：

$$E_{\mathrm{LJ}} = 4\varepsilon \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \left[\left(\frac{\sigma}{r_{ij}}\right)^{12} - \left(\frac{\sigma}{r_{ij}}\right)^{6} \right] \tag{6}$$

其中，N表示原子个数，此处为20，r_{ij}表示原子i与原子j之间的距离，$\varepsilon = 1.0$，$\sigma = 1.0$。

构成物体的分子的能量由两部分构成——分子动能和势能。我们欲验证团簇的能量大部分是由势能提供，由于量纲差异，无法确定其准确性，但可由其波动性确认势能与能量波动一致。

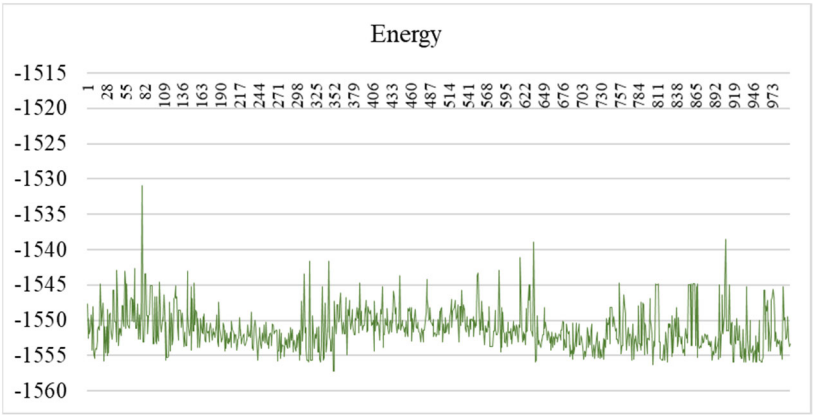


图 1 Au_{20}的能量波动图

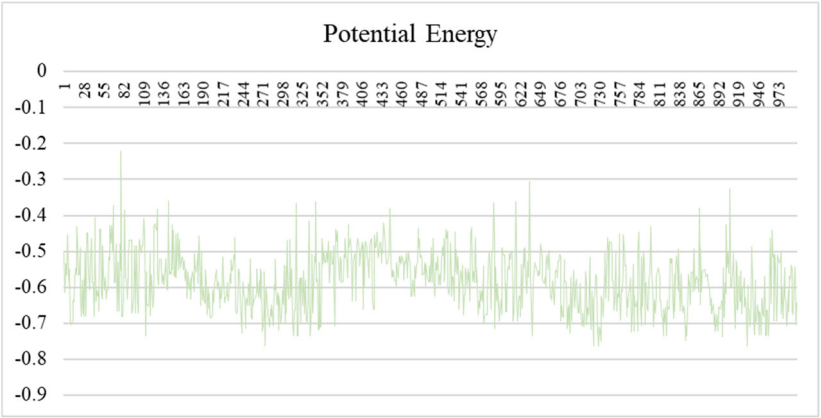


图 2 Au_{20}的势能波动图

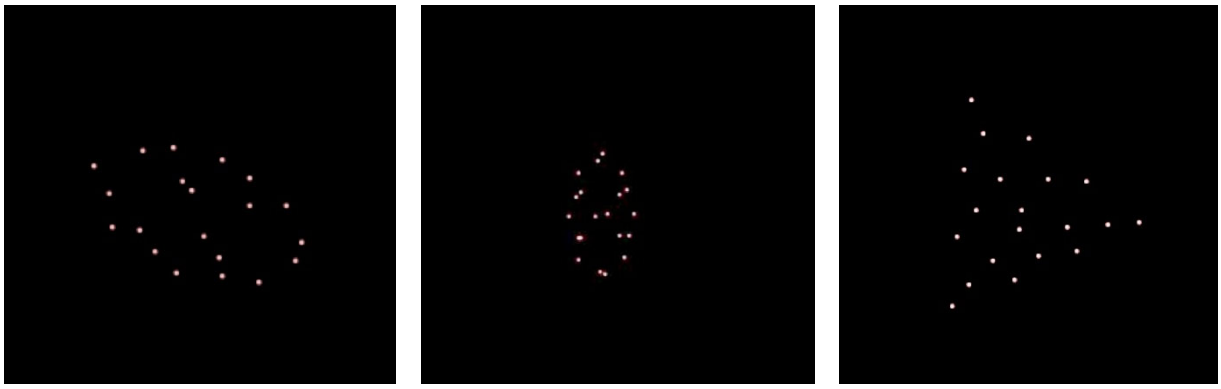
由图 1、图 2 可知，Au_{20}的势能与其能量波动情况一致，均在第 78 个文件所指的结构时达到峰值，在 energy 为-1530.908363 时，势能为-0.221525215463521。在第 350 个文件所指的结构时达到谷值，在 energy 为-1557.20946 时，势能为-0.764308644879803。

后续我们将势能和题给能量作为因变量进行测试，增加结果的可信度。

4.3 模型的求解

4.3.1 Au_{20}的结构探索

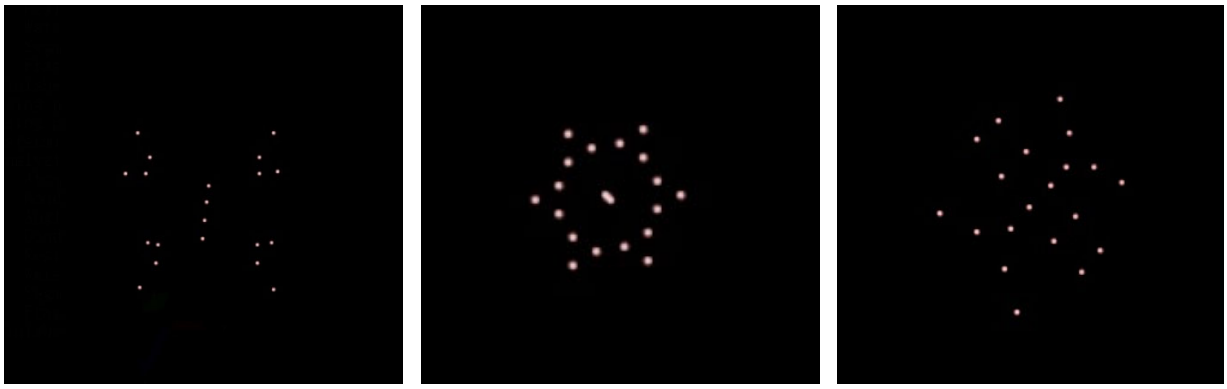
我们对能量值排列前后二十位的.xyz 文件通过 VMD 进行可视化，得到一些结果，如下图。



第 315 号　　第 958 号　　第 351 号

图 3 稳定构型的 VMD 可视化

图 4 稳定构型的样例



第 78 号　　第 317 号　　第 586 号

图 5 非稳定构型的 VMD 可视化

图 6 非稳定构型的样例

我们发现，第 315 号和第 958 号与图 4 的样例相似度较高，第 317 号和第 586 号贴近于平面图形，与图 6 的样例重合度较高，第 351 号是所给数据中能量最低对应的最优构型，第 78 号是所给数据中能量最高对应的最差构型。

我们猜想，构型单支数越多、连接度越弱、立体感越差的构型稳定性越差。

4.3.2 特征值的选取与预测模型的构建

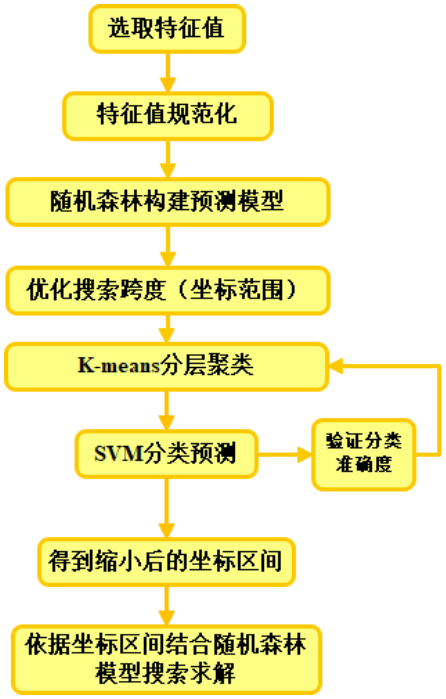


图 7 模型构建流程图

我们欲采用随机森林算法或 BP 神经网络建立Au_{20}能量预测模型。选取 20 组三维坐标、原子密度、团簇体积作为输入集，能量值、势能值作为输出集。选用全体数据的 90%作为训练集，另 10%作为预测集，验证、比对得到最佳模型。

采用随机森林法，是因为它对多元共线性不敏感（对于多元回归中特征值粘黏性的数据集影响较小）。我们无法确定坐标之间是否有粘黏性，因为我们推测稳健的构型是对称的，所以坐标之间一定是有关联的。此外，随机森林对缺失数据和非平衡数据比较稳健，可以很好地预测多达几千个解释变量的作用。我们不采用 LSTM 等单一特征值算法，是因为我们输入的特征值多达 62 个，而随机森林可以很好的解决这个问题，并且能避免陷入过拟合的情况。

采用 BP 神经网络法，是因为已经被证实，三层 BP 神经网络能够逼近任何有理函数。由于我们的内部映射机制比较复杂，BP 神经网络可以解决任何复杂非线性映射。

我们采用两种算法，得到了如表 1 的误差结果。

评价指标	MSE	RMSE	MAE	R^2	MAPE
随机森林	0.2997	0.5475	0.3134	0.9637	0.0202
BP 神经网络	240638…	155125…	155124…	-291321.1646	100.0000

表 1 测试效果对比

由上表看出，随机森林的 R^2 逼近 1，且误差值贴近 0，预测效果较好。而 BP 误差较大，结果极度不合理，我们猜测它是陷入了局部极小化的问题，在初始化网络的时候参数设定错误，使得其过度局部收敛，陷入僵局。同时，BP 输入的自变量意义度较低，均导致失败。故我们采用随机森林构建模型。

4.3.3 K-means 聚类与 SVM 分类实现

K-means 算法取输出参数k，将一组n个对象分成k个簇，使同一簇内的相似性最高，不同簇间的相似性较低。我们将 K-means 使用的点集变成$\left(E_{总}, Ep\right)$，我们对E_p扩充量级，使得它与$E_{总}$处于同一量级，经过测试，我们发现$K = 4$聚类效果最好。如图 7 所示，数据集被很好地切割成 4 个部分。

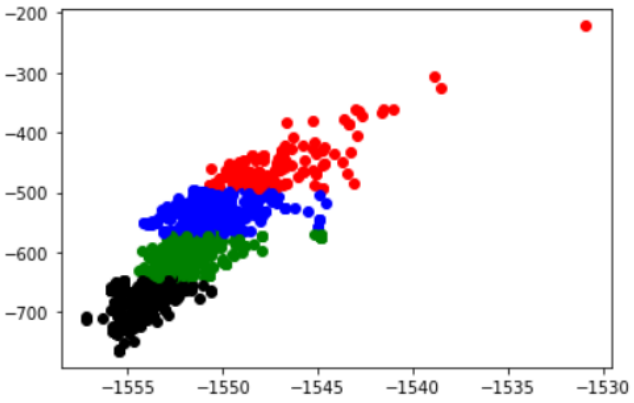


图 8 K-means 聚类

我们采用 K-means 聚类是为了找到最优坐标的区间，即对于 20 个坐标分别估计其合适区间，缩小搜索跨度，并能进一步减小步长，提高搜索精度。为了使 K-means 分类效果最高，我们同时采用 SVM 分类器进行验证。SVM 的优点是针对于小样本进行训练，可以实现高维度的非线性映射，正好适用于我们的数据集，分类效果较好。SVM 的最终准确率为 81.5%。

4.3.4 最优坐标区间的选取与数据集准备

我们对 K-means 聚类得到的第二类（能量最低）进行描述性统计。

描述统计

	N	范围	最小值	最大值	合计	均值		标准 偏差	方差	偏度		峰度	
	统计	统计	统计	统计	统计	统计	标准 错误	统计	统计	统计	标准 错误	统计	标准 错误
Energy	213	2.740686	-1557.209460	-1554.468774	-331291.618	-1555.35971	.042605498	.621806776	.387	-.906	.167	1.403	.332
X1	213	6.65361774	-3.45053383	3.20308391	22.93648989	.1076830511	.0827738461	1.208044513	1.459	-.136	.167	2.009	.332
Y1	213	7.24634007	-3.28538130	3.96095877	-.00973288	-.0000456943	.0708261593	1.033673764	1.068	.425	.167	5.562	.332
Z1	213	8.88812105	-5.21001501	3.67810604	-11.58645364	-.0543964960	.1509614321	2.203209568	4.854	-.394	.167	-.778	.332
X2	213	7.73111404	-4.28058025	3.45053379	-20.27857012	-.0952045546	.0838363958	1.223551915	1.497	-.534	.167	2.916	.332
Y2	213	7.24634007	-3.96095877	3.28538130	-11.29855076	-.0530448392	.0657528431	.9596311526	.921	-.484	.167	7.191	.332
Z2	213	8.88811664	-3.67810604	5.21001060	12.09297054	.0567745096	.1573198905	2.296008213	5.272	.386	.167	-.820	.332
X3	213	7.49913892	-3.54649127	3.95264765	49.01099157	.2300985520	.1484452050	2.166486442	4.694	.017	.167	-1.291	.332
Y3	213	7.39612427	-3.30844042	4.08768385	39.91421069	.1873906605	.0962970800	1.405409613	1.975	.329	.167	.522	.332
Z3	213	7.79630511	-4.31193637	3.48436874	-102.968752	-.4834213726	.1162425473	1.696504125	2.878	.147	.167	.074	.332
X4	213	7.41479355	-3.97526122	3.43953233	-96.22881663	-.4517784818	.1125480646	1.642584925	2.698	-.073	.167	.030	.332
Y4	213	7.39629632	-4.08785589	3.30844043	6.99396148	.0328354999	.1362748529	1.988866001	3.956	-.167	.167	-1.243	.332
Z4	213	8.59316393	-4.22485791	4.36830602	24.09213112	.1131085968	.1270773912	1.854633467	3.440	.090	.167	-.196	.332
X5	213	8.18319297	-4.64300592	3.54018705	19.23164090	.0902893939	.1295343280	1.890491279	3.574	.019	.167	-.738	.332
Y5	213	7.38634325	-3.49577216	3.89057109	25.29805928	.1187702314	.1338945761	1.954127005	3.819	.136	.167	-1.005	.332
Z5	213	7.53586709	-4.22944029	3.30642680	17.34603610	.0814367892	.1057753403	1.543740268	2.383	-.333	.167	.576	.332
X6	213	7.82903018	-3.18602427	4.64300591	-24.51346213	-.1150866767	.1357697156	1.981493764	3.926	.224	.167	-.899	.332
Y6	213	7.58345665	-4.08768385	3.49577280	-127.576055	-.5989486132	.1270442655	1.854150013	3.438	.016	.167	-.625	.332
Z6	213	6.53545083	-3.22901852	3.30643231	-1.99255801	-.0093547324	.0949248841	1.385383074	1.919	.204	.167	.206	.332

图 9 描述性统计

第二类共有 213 个数据，我们成功将x, y, z的坐标区间由原本的$[-8,8]$缩小为现在的$[-3,3]$。接下来，我们利用 RAND()函数随机生成 100 万条数据（坐标包含在缩小后的区间内），准备批量导入随机森林预测模型。

4.3.5 模型结果分析

我们将 100 万条数据分成 10 次导入模型，选取预测结果最佳的 4 条数据进行构图。其预测值与最低能量极为贴切，最低能量为-1557.20946。

X	Y	Z	X	Y	Z
1.032969	-1.06001	1.036414	-0.9886	0.929053	0.977141
-0.96369	-1.00063	-0.88549	0.927707	0.869489	-0.98307
-0.98718	-1.00144	2.927839	0.90583	0.899616	2.756109
-0.82021	0.965641	-2.84768	0.820528	-0.87931	-2.82381
0.972615	2.90605	0.911178	-1.04825	-2.91271	0.856655
2.793802	-1.02891	-0.98656	-2.80378	0.939459	-0.83632
-0.8578	2.782448	-0.88898	0.868657	-2.83006	-0.87043
-2.79407	-2.71378	2.819605	2.860111	2.621133	2.688232

表 2 最优坐标解

作三维图，如下图所示。观察为类正四面体，验证猜想。

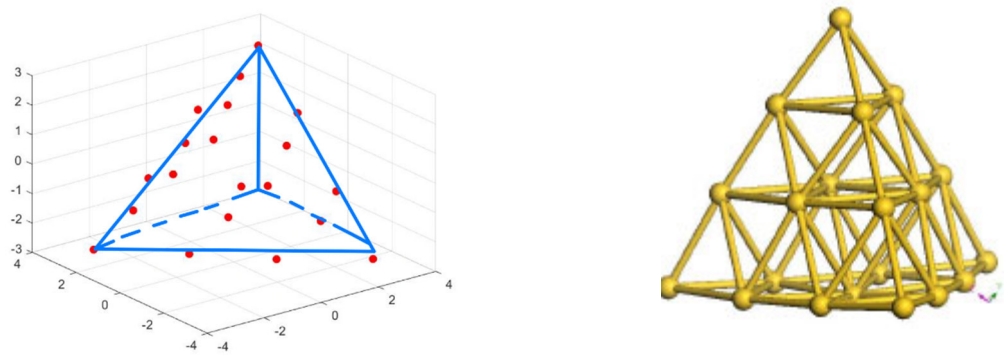


图 10 最优结构图

4.4 模型的改进

通过阅读大量文献，我们了解到求解全局最优结构的算法可分为有偏算法和无偏算法。无偏算法是指从随机生成的构型开始，基于随机扰动的方式进行优化操作，使用该算法能够找到实例的最低能量构型。本文选择的随机生成构型的方法，导致算法的计算速度受限、成功率尚可。为了进一步发展团簇的结构优化算法，我们期待使用无偏优化算法中的 DLS-TPIO 算法来有效地提高算法的效率。在 DLS-TPIO 算法中,内部操作、两阶段局部搜索、动态格点搜索方法分别起了重要的作用。在优化的前一阶段,内部操作将某些能量较高的表面原子转移至团簇的内部, 从而降低团簇的能量。同时,两阶段局部搜索方法引导搜索进入可能性更高的构型区域。这种方式极大地提高了算法的成功率。在优化的后一阶段,该算法使用动态格点搜索方法对表面原子的位置作进一步优化,再一次降低了团簇的能量。

随着研究的逐渐深入, 我们也希望将 DLS-TPIO 算法应用于其他原子团簇的结构优化。

五、问题二的模型建立与求解

针对问题二，我们首先查阅资料，Au_{20}、Au_{32}、Au_{58}等具有较高的稳定性。研究表明，原子数小于 10 的Au_n呈平面结构，在$n = 19$时会向金字塔结构转变，在$n = 20$左右呈金字塔结构，在$n = 29{\sim}32$范围内呈现为类富勒烯结构。下图为团簇稳定性与原子数的关系示意图。

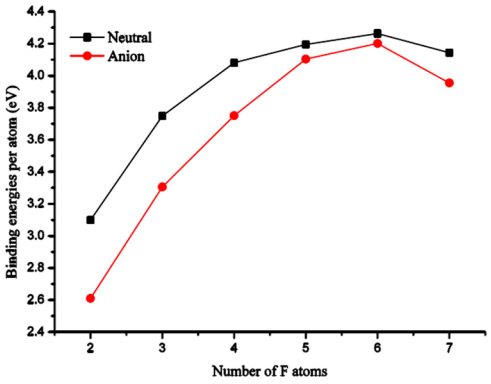


图 11 团簇稳定性与原子数的关系

5.1 正四面体的拓展模型建立

由问题 1 可得，稳定结构为正四面体。我们推测Au_{32}的整体结构可拆分为多个正四面体。Au_{32}比Au_{20}多了 12 组三维坐标，我们猜想，分别由四面体的四个表面，由表面中心作该面垂线，在垂线上找寻一点，成为Au_{32}的点集中的一员。做完一轮可以扩充四个点，我们需要扩充 12 个点，因此可以在新生成的面上继续重复操作，以此类推，得到多种异构体，并希望在其中探寻最优结构体。

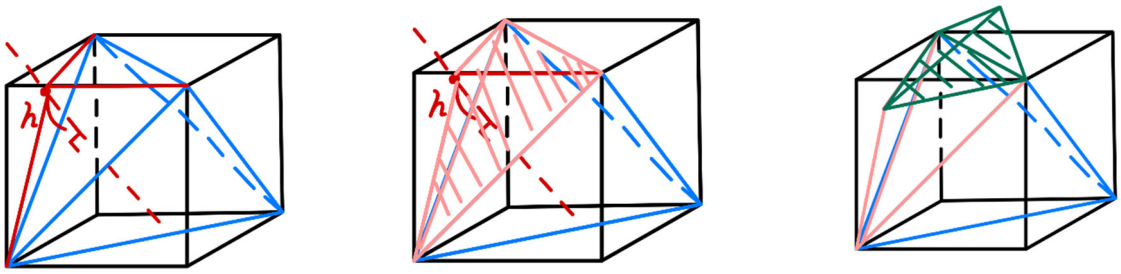


图 12 正四面体拓展方式

由于三角形是最稳定的，我们希望每个面扩充后仍为三角形，因此，我们重复扩充四面体的操作，希望这样可以探寻到最优解。

5.2 模型的求解

问题一中我们建立了 62 输入—2 输出的模型，而现在输入量变为 96+2，因此，我们进行循环操作，每次取 96 组坐标中的 60 组，希望可以一直输出四面体的构型并存储能量预测值，使预测值始终较低水平。同样类似问题一，我们选取合适的区间，扩充 100 万次数据，找到了一些不规则的异构体，以下为异构体的图像。

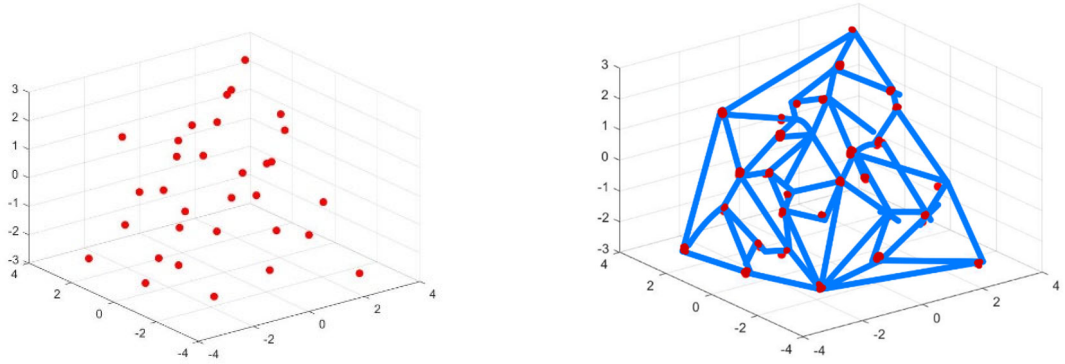


图 13 Au_{32}异构体

由于模型较为简陋，输出效果不佳，但我们推测Au_{32}的最优结构体应当类似足球结构，为多个正六边形或三角形组成的多面体。以下放入理想构图：

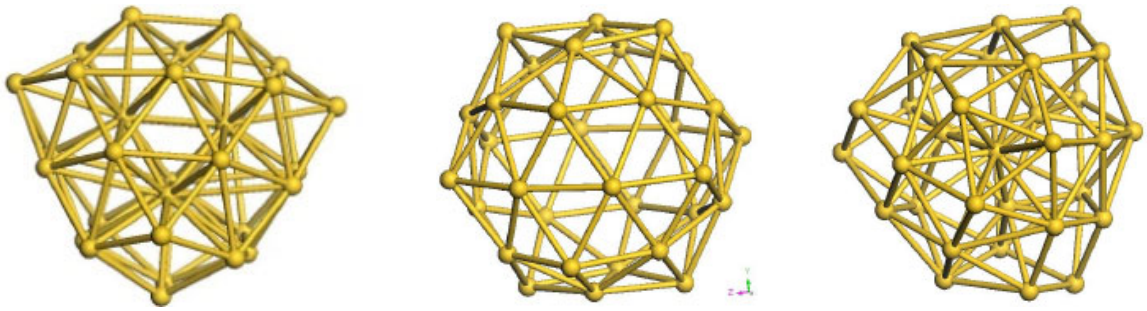


图 14 Au_{32}较优结构体

六、问题三的模型建立与求解

仿照问题一，我们发现二者差异仅在原子属性、电负性上。大体结构仍仿照问题一求解。

6.1 数据预处理

通过分析比对，我们同样分别对 3751 个坐标数据、能量数据、原子名称进行分离，得到了 3751×137 的数据表单。对能量值进行画图，得到能量值域。取前十与后十个数据，载入 VMD 进行直观分析，对硼团簇的构型有初步认知。

相较问题一，硼团簇应当引入电子相关能。电子之间存在库仑排斥，不能独立运动，每个电子在自己的周围建立了一个“库伦穴”，降低其他电子进入的概率。但事实上，电子相关能在体系中的占比并不大，约为 0.3%~1%。因此，我们可以忽略这种影响。

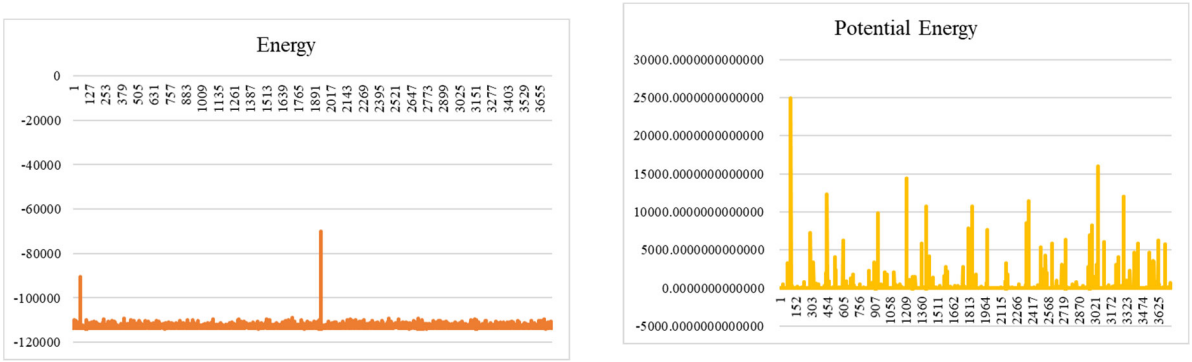
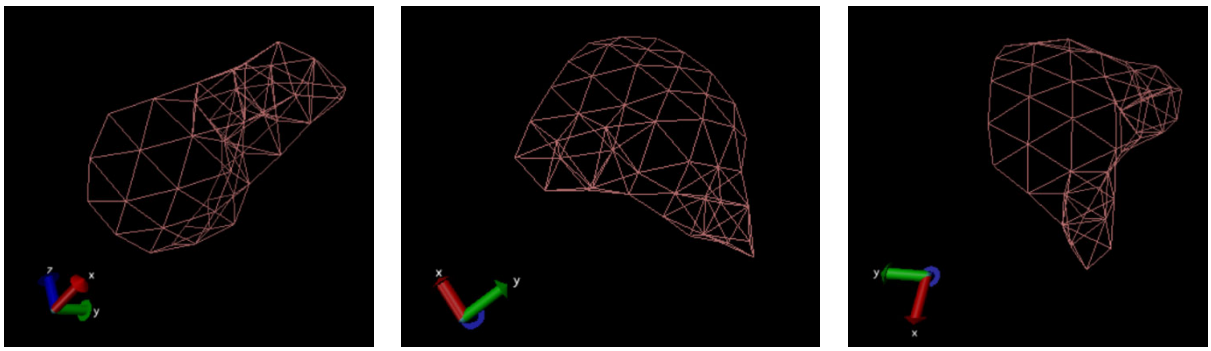


图 15 硼团簇能量—势能对比图

我们发现，对比于金团簇，硼团簇二者相差较大。我们猜想，除去势能，电子的动能对于该结果的影响较大，由于题给参数不足，我们无法估测电子的动能，故舍去势能。

6.2 模型的建立与求解

我们对能量值排列前后二十位的.xyz 文件通过 VMD 进行可视化，得到一些结果，如下图。



第 273 号　　第 3488 号　　第 3470 号

图 16 稳定构型的 VMD 可视化

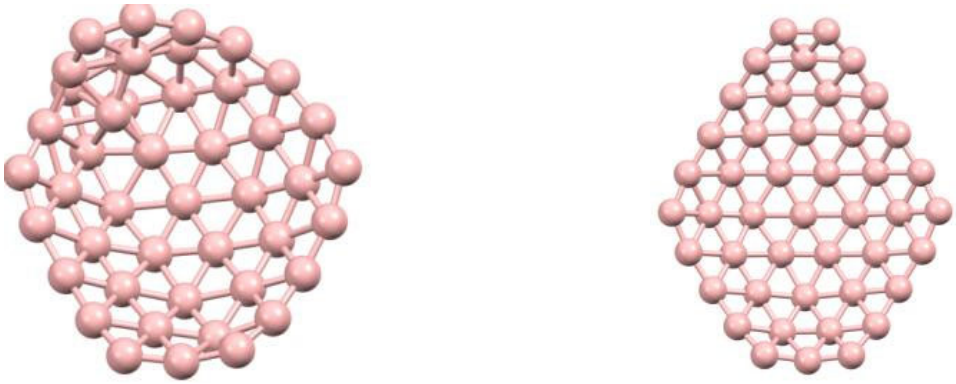
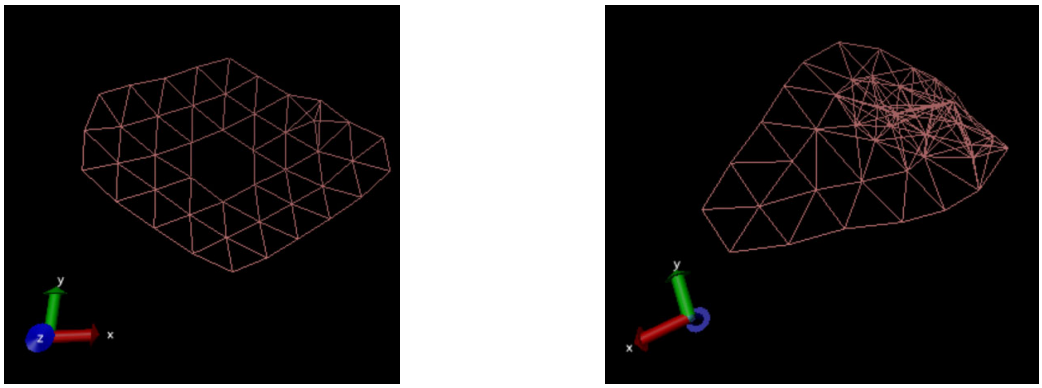


图 17 稳定构型的样例



第 2746 号　　第 105 号

图 18 非稳定构型的 VMD 可视化

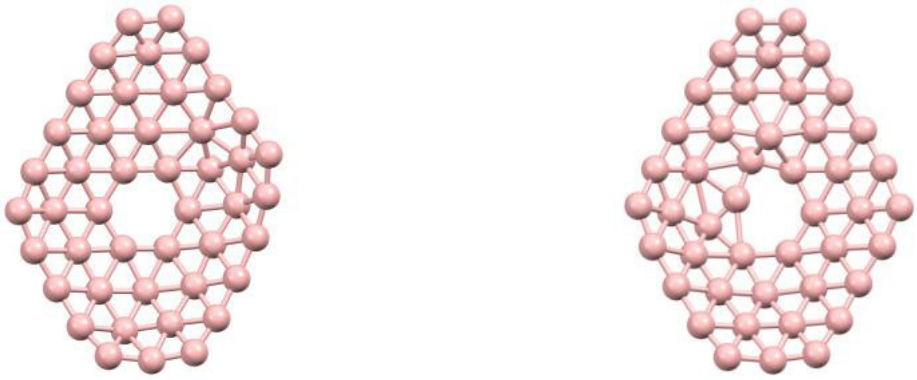


图 19 非稳定构型的样例

我们发现，第 273 号和第 3488 号与图 15 的样例相似度较高，第 2746 号和第 105 号贴近于平面图形，与图 17 的样例重合度较高，第 273 号是所给数据中能量最低对应的最优构型，第 2746 号是所给数据中能量最高对应的最差构型。

我们猜想，呈笼状结构、无缺口、立体感越好的构型稳定性越好。

仿照问题一，我们同样进行 K-means 聚类，并以$K = 4$为最佳分类结果。依据结果，我们进行数据扩充，同样扩充至 100 万条，准备批量导入随机森林预测模型。

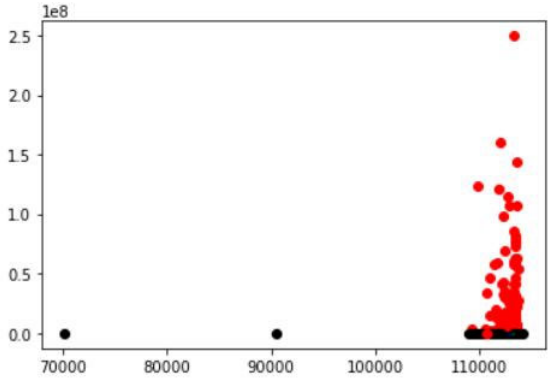


图 30 K-means 聚类

我们将 100 万条数据分成 10 次导入模型，选取预测结果最佳的 6 条数据进行构图。其预测值与最低能量极为贴切，最低能量为-113946.567430971。

X	Y	Z
1.654844227	0.051877268	0.046985998
1.872800069	-1.912661439	-0.357629898
0.524717562	0.811565758	-0.140634709
-0.700720138	2.422807206	0.185242188
0.647158448	0.103044684	-0.096928558
-1.042741456	1.610892512	-1.868209911
-2.178610254	0.737882018	0.548381411
3.143530274	0.442988409	0.455917537
-1.462937498	1.971429787	0.188885546
3.308450073	-2.014165114	1.174318168
-1.161030792	3.887579207	-0.428305378
-3.069408751	-1.227435834	0.456302376
-3.174556231	0.90276027	-0.750356501
-3.920902346	-0.752117851	-0.103923758
0.26446033	-0.823308359	-0.386955292
-1.626217419	-1.999895516	0.360313173
-3.158693688	-1.006896194	0.503343569
-4.133557922	-0.564603594	0.974602173
-3.822304926	-1.978143411	1.26744404
-2.250139183	-2.282558922	0.661546209
-1.009572797	-3.486453952	0.542007611
-0.216801661	-3.013144927	0.615624792
-0.000218341	-3.88597359	0.753005352
2.459385772	-2.942040693	0.380307297
1.343573255	-3.668582337	0.486548016
2.942170366	-1.337492411	0.05720938
4.297708917	-2.525168748	0.659507804
3.758890234	-0.946180391	-0.110114507
1.261656694	-2.304379265	-0.111945266
0.997902214	1.921722276	-0.460630916

-2.768829021	-1.127592999	-0.688041126
1.57498224	-0.161583775	-0.760109257
-4.274944925	-1.547028551	0.463599447
5.351363641	-0.079879405	1.032268896
5.151842812	-0.57751979	0.661839196
5.078087396	-1.171088692	0.674129617
6.115617342	-0.640570887	0.60903108
6.105179948	-0.619064693	0.423190593
-1.139510448	-0.242612538	-0.472601601
3.034779822	0.695632405	-0.328340717
-0.223829226	0.5938022	-0.328380783
-4.517087209	-0.712214191	1.202721582
-5.843813568	-0.275267964	2.025996145
-4.97614701	-0.58886126	1.446525643
0.695144192	0.375805023	-0.353275799

表 3 最优坐标解

作三维图，如下图所示。观察为较复杂的折叠构型，验证猜想。

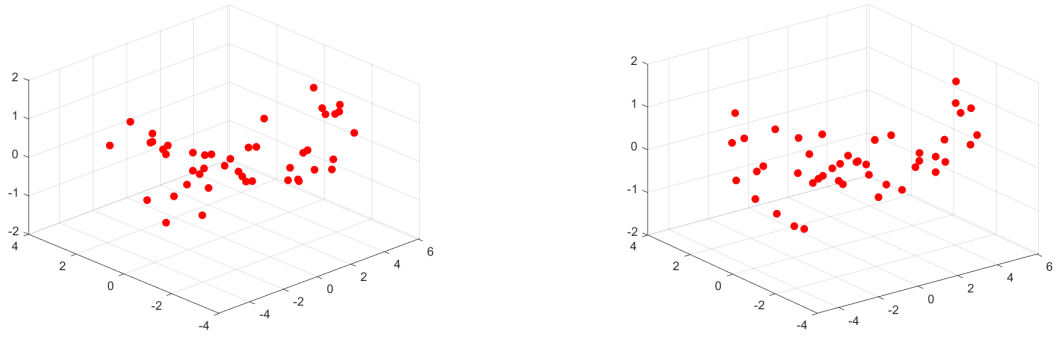


图 20 最优结构图

七、问题四的模型建立与求解

7.1 数据预处理

仿照问题二，现在题中要求输入 40 个坐标，我们欲设置 5 个无关坐标，仍保持输入集为 45 进行预测。

7.2 模型的建立与求解

我们通过预测值，通过排序找到能量最低的结构坐标，并以三维图呈现。

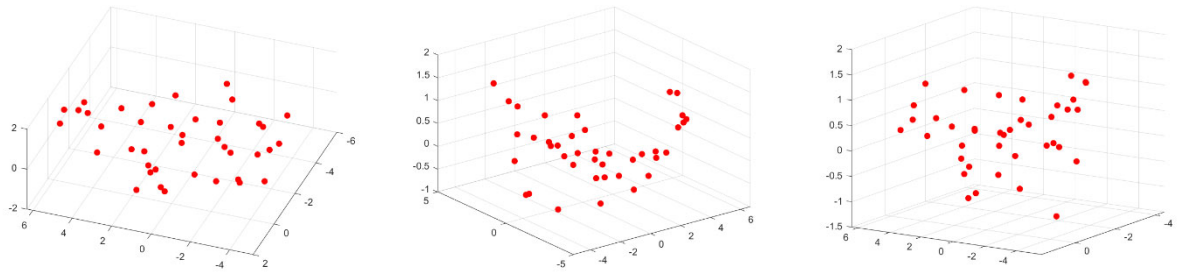


图 21 最优结构图

经过查阅资料，我们得知B_{40}^{-}团簇有两个共存的全局最优结构：其一为准平面构型并有C_S对称性，另一则是较为罕见的具有D_{2d}对称性的三维高对称立体结构。

八、模型的优缺点

8.1 模型的优点

采用随机森林算法，经检验为预测模型中兼容性较好的算法，对于数据的要求度不高，使得预测结果较为理想。

8.2 模型的缺点

采用的搜索算法为随机模拟与暴力枚举、耗时费力、可以采用梯度下降的方法进行优化、或者对搜索结果进行剪枝处理。此外，应当去考虑结合团簇构型的空间搜索算法得到坐标最优解，更符合化学特征。

参考文献

[1]高锦花. 金纳米团簇的结构及其构效关系研究[D].北京理工大学,2016.

[2]赖向京,许如初,黄文奇.Lennard-Jones 团簇最低能量构型的预测[J].中国科学:化学,2011,41(07):1137-1144.

[3]姚文志. 硼氧及硼金团簇结构与性质的理论研究[D].山西大学,2010.

附录

```
import pandas as pd
import numpy as np
import os
import csv
import math

#Au 的势能和原子密度
Potential_Energy = np.zeros(999)
Atomic_Density=np.zeros(999)
for k in range(1,1000):
    for i in range((k-1)*20,k*20-1):
        for j in range(i+1,k*20):

euclidean_temp=(X[i]-X[j])*(X[i]-X[j])+(Y[i]-Y[j])*(Y[i]-Y[j])+(Z[i]-Z[j])*(Z[i]-Z[j])
                euclidean_result=pow(euclidean_temp,0.5)
                if(euclidean_result==0):
                    print(i,j)
                euclidean_result=pow(euclidean_result,-1)
                temp1=pow(euclidean_result,12)
                temp2=pow(euclidean_result,6)
                t=temp1-temp2
                temp3=pow(euclidean_result,3)
                Potential_Energy[k-1]+=t
                Atomic_Density[k-1]+=temp3
    Potential_Energy[k-1]=Potential_Energy[k-1]*4

#Au 的体积
max_x=np.zeros(999)
max_y=np.zeros(999)
max_z=np.zeros(999)
min_x=np.zeros(999)
min_y=np.zeros(999)
```

```
min_z=np.zeros(999)
volumn_result=np.zeros(999)
flag=0
for k in range(1,1000):
    for i in range((k-1)*20,k*20):
        if(max_x[k-1]<X[i] or flag==0):
            max_x[k-1]=X[i]
        if(max_y[k-1]<Y[i] or flag==0):
            max_y[k-1]=Y[i]
        if(max_z[k-1]<Z[i] or flag==0):
            max_z[k-1]=Z[i]
        if(min_x[k-1]>X[i] or flag==0):
            min_x[k-1]=X[i]
        if(min_y[k-1]>Y[i] or flag==0):
            min_y[k-1]=Y[i]
        if(min_z[k-1]>Z[i] or flag==0):
            min_z[k-1]=Z[i]
        flag=1
    flag=0

volumn_result[k-1]=(max_x[k-1]-min_x[k-1])*(max_y[k-1]-min_y[k-1])*(max_z[k-1]-min_z
[k-1])


#B_kmeans
from sklearn.cluster import KMeans
from sklearn import preprocessing
import pandas as pd
import numpy as np
import os
kmeans = KMeans(n_clusters=4)
# 归一化
min_max_scaler=preprocessing.MinMaxScaler()
train_x=min_max_scaler.fit_transform(train_x)
# kmeans 算法
```

```
kmeans.fit(train_x)
predict_y = kmeans.predict(train_x)
#  合并聚类结果
result = pd.concat((data,pd.DataFrame(predict_y)),axis=1)
result.rename({0:u'聚类'},axis=1,inplace=True)
result.to_csv('B_kmeans_4.csv')



#SVM 分类器
from sklearn import svm
from sklearn import metrics
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScal

train,test = train_test_split(data,test_size = 0.2)
#数据规范化
standard_scaler = StandardScaler()
train_x = standard_scaler.fit_transform(train_x)
test_x = standard_scaler.fit_transform(test_x)
#创建 SVM 分类器
model = svm.SVC()
#训练数据
model.fit(train_X,train_y.astype('int'))
#模型评估
prediction = model.predict(test_x)
print('准确率:',metrics.accuracy_score(prediction,test_y))
```