中国研究生创新实践系列大赛

“华为杯”第十八届中国研究生

数学建模竞赛

学　　校	东北大学
参赛队号	**21101450003**
队员姓名	**1.杨少雄**
	2.牟宝奎
	3.董建荣

中国研究生创新实践系列大赛

“华为杯”第十八届中国研究生

数学建模竞赛

题 目 抗乳腺癌候选药物的优化模型

摘 要：

乳腺癌是目前世界上最常见，致死率较高的癌症之一，故寻找出最适合抗乳腺癌候选药物，具有重要意义。本文依次进行筛选变量、相关分析、定量预测、分类预测操作，对题目所给的问题进行快速有效求解，整体构建出抗乳腺癌候选药物优化模型。

针对问题一，对所给 1974 个化合物的 729 个分子描述符原始数据进行**预处理**以及确定了前 20 个对生物活性最具有显著影响的分子描述符并给出变量对生物活性影响的重要排序表。首先，选取了代表生物活性的量级相差较小的指标 pIC_{50}，再利用数据处理手段删除掉**无关变量**；再次，在剩余数据集中，根据**皮尔森相关系数**过滤掉与生物活性相关性较低（相关系数小于 0.4）的变量实现第一级变量降维，得到相关性较强的 24 个分子描述符；然后，为了更好的描述分子描述符对生物活性的显著影响，采用**灰色关联分析**，求出生物活性与分子描述符之间的**关联度系数**，筛除掉关联度系数小于 0.95 的变量，得到 20 个分子描述符，实现第二级变量降维；最后，广泛查阅相关文献，综合确定了前 20 个对生物活性最具有显著影响的分子描述符并给出变量对生物活性影响的重要排序表。

针对问题二，建立化合物对 ER 生物活性的定量预测模型。首先，基于问题二中挑选的前 20 个对生物活性最具有显著影响的分子描述符，对变量数据进行**散点图分析**，判断出描述符变量与代表 ERα生物活性的指标 pIC_{50} 之间具有高度的非线性关系。然后，基于数据的非线性关系，利用描述符变量数据和**多元线性回归、双隐层 BP 神经网络、随机森林算法**进行定量预测代表 ERα生物活性指标 pIC_{50} 值，并对三种模型效果进行分析得到：多元线性回归预测模型的**均方误差** MSE 为 1.584，**平均绝对百分比误差** MAPE 为 35.69% ；双隐层 BP 神经网络预测模型的均方误差 MSE 为 0.929，平均绝对百分比误差 MAPE 为 17.46% ；随机森林预测模型的均方误差为 MSE 为 0.287，平均绝对百分比误差 MAPE 为 3.71% 。最后，根据横向对比三种预测模型的预测的准确性以及鲁棒性，确定化合物对 ERα生物活性的**随机森林定量预测模型**并利用训练的预测模型对 50 个化合物进行对应的 pIC_{50} 值预测，再利用 IC_{50} 值与 pIC_{50} 值关系转换公式计算出 IC_{50} 值，并将结果分别填入表中的 IC50_nM 列及对应的 pIC50 列。

针对问题三，建立构建化合物的 Caco-2、CYP3A4、hERG、HOB、MN 的分类预测模型。首先，依据提供的 729 个分子描述符以及 1974 个化合物的 ADMET 数据，判断出 Caco-2、CYP3A4、hERG、HOB、MN 为**二值类别变量**。

然后，基于二值类别变量的类型，利用随机森林中的基尼指数选择最优特征的方法分别计算出影响 Caco-2、CYP3A4、hERG、HOB、MN 的特征重要性，并筛选出特征重要性大于 0.008 的分子描述符。其次，将选出的分子特征符作为自变量，分别用**随机森林二分类、支持向量机二分类算法**（SVM）对化合物的 Caco-2、CYP3A4、hERG、HOB、MN 的建立分类预测模型。再次，对随机森林二分类、支持向量机二分类模型预测效果进行分析得到：支持向量机二分类预测模型对 Caco-2、CYP3A4、hERG、HOB、MN 的**预测准确率**分别为 85.81%、89.31%、89.80%、91.85%、85.29%；随机森林二分类预测模型对 Caco-2、CYP3A4、hERG、HOB、MN 的预测准确率分别为 90.91%、90.68%、92.52%、81.11%、91.83%。最后，通过对比二种模型预测效果，对于 Caco-2、CYP3A4、hERG、MN 选取随机森林二分类预测模型，对于 HOB 选取支持向量机二分类预测模型，对文件表中的 50 个化合物进行相应的预测，并将结果填入表中对应的 Caco-2、CYP3A4、hERG、HOB、MN 列。

针对问题四，建立分子描述符**最优范围优化**模型。首先，查阅到美国 NCI 建议指出，IC50 大于 50uM 可认为该化合物生物活性几乎没有，即 pIC50 需要大于 5.3。其次，挑选出满足 pIC50 值大于 5.3 条件的化合物，得到 979 个样本数据。再次，利用假设，将 MN、hERG 取值为 0，筛选化合物，进一步得到 236 个样本数据。然后，取问题一与问题二中建立的六个模型中的对模型有显著影响分子描述符，取六个模型自变量的并集，选出了对六个模型有影响的 70 个分子描述符。进一步，取 Caco-2、CYP3A、HOB 共有 7 个可能取值组合并集，可以得到 97 个样本。最后，对于连续变量，寻找分子描述符的$(Q_1-1.5IQR, Q_3+1.5IQR)$作为取值范围；对于非连续变量，寻找出现次数最多的值作为最优取值。

最后，对问题的模型与算法中优缺点进行总结。

关键词：皮尔森相关系数； 灰色关联分析； 多元线性回归； 双隐层 BP 神经网络； 随机森林； 支持向量机

目录

1 问题背景与问题重述

1.1 问题背景

乳腺癌是目前世界上最常见，致死率较高的癌症之一。乳腺癌对于女性来说也是最常见的恶性肿瘤之一。研究表明，ERα确实在乳腺发育过程中扮演了十分重要的角色，并且 ERα被认为是治疗乳腺癌的重要靶标。因此，充分利用能够拮抗 ERα活性的化合物可能是治疗乳腺癌的候选药物依据，建立抗乳腺癌候选药物优化模型，对高效治疗乳腺癌具有重大意义。

在药物研发中，通常采用建立化合物活性预测模型的方法来筛选潜在活性化合物。一个化合物想要成为候选药物，除了需要具备良好的生物活性（此处指抗乳腺癌活性）外，还需要在人体内具备良好的药代动力学性质和安全性，合称为 ADMET（Absorption 吸收、Distribution 分布、Metabolism 代谢、Excretion 排泄、Toxicity 毒性）性质。一个化合物的活性再好，如果其 ADMET 性质不佳，比如很难被人体吸收，或者体内代谢速度太快，或者具有某种毒性，那么其仍然难以成为药物，因而还需要进行 ADMET 性质优化。

首先需要构建化合物生物活性的定量预测模型和 ADMET 性质的分类预测模型，从而为同时优化 ERα拮抗剂的生物活性和 ADMET 性质提供预测服务，才能更好的建立抗乳腺癌候选药物优化模型，解决乳腺癌药物挑选难问题。

1.2 问题重述

基于上述研究背景，本文需要解决以下问题：

1.2.1 问题一：数据的预处理以及寻找建模主要变量

根据文件“Molecular_Descriptor.xlsx”和“ERα_activity.xlsx”提供的数据，针对 1974 个化合物的 729 个分子描述符进行变量选择，根据变量对生物活性影响的重要性进行排序，并给出前 20 个对生物活性最具有显著影响的分子描述符（即变量），并请详细说明分子描述符筛选过程及其合理性。

1.2.2 问题二：建立定量预测模型

请结合问题 1，选择不超过 20 个分子描述符变量，构建化合物对 ERα 生物活性的定量预测模型，请叙述建模过程。然后使用构建的预测模型，对文件“ERα_activity.xlsx”的 test 表中的 50 个化合物进行 IC50 值和对应的 pIC50 值预测，并将结果分别填入“ERα_activity.xlsx”的 test 表中的 IC50_nM 列及对应的 pIC50 列。

1.2.3 问题三：建立分类预测模型

请利用文件“Molecular_Descriptor.xlsx”提供的 729 个分子描述符，针对文件“ADMET.xlsx”中提供的 1974 个化合物的 ADMET 数据，分别构建化合物的 Caco-2、CYP3A4、hERG、HOB、MN 的分类预测模型，并简要叙述建模过程。然后使用所构建的 5 个分类预测模型，对文件“ADMET.xlsx”的 test 表中的 50 个化合物进行相应的预测，并将结果填入“ADMET.xlsx”的 test 表中对应的 Caco-2、CYP3A4、hERG、HOB、MN 列。

1.2.4 问题四：抗乳腺癌候选药物优化模型

寻找并阐述化合物的哪些分子描述符，以及这些分子描述符在什么取值或者处于什么取值范围时，能够使化合物对抑制 ERα 具有更好的生物活性，同时具有更好的 ADMET 性质（给定的五个 ADMET 性质中，至少三个性质较好）。

2 模型的假设与符号说明

2.1 模型假设

根据所给题目信息以及要求，本文作如下假设：
（1）假设题目中附件所提供的数据是真实可行的；
（2）假设做问题四前，前三问所建立的模型挑出的分子描述符合理；
（3）假设 hERG 取值 1 时，代表的该化合物心脏毒性，具有致命的毒性；
（4）假设 MN 取值为 1 时，代表的该化合物具有遗传毒性，认为后代一定会具有该遗传毒性；

2.2 符号说明

表 2.1 符号说明

序号	符号	含义
1	r	Pearson 相关系数
2	λ	惩罚系数
3	T	随机森林中树的总数
4	N	样本数
5	Q_3	下四分位数
6	Q_1	上四分位数

3 问题一的分析与求解

3.1 问题分析

针对问题一，题目要求根据分子描述符（变量）对生物活性影响，将变量对生物活性影响重要性进行排序，并给出前 20 个对生物活性最具有显著影响的分子描述符。

首先，由于问题研究的因变量是生物活性，从而需要选取出代表生物活性的指标，通过观察所给的化合物对 ERα的生物活性指标 IC_{50} 值，pIC_{50} 值，发现 IC_{50} 数值波动比较大，考虑到模型建立的鲁棒性，从而采用量级相差较小的指标 pIC_{50} 值来代表化合物对 ERα的生物活性。

其次，鉴于所给的数据集比较大，需要采取了一些预处理手段，删除掉了无关变量。无关变量的剔除原则为：无论有没有该个变量，对因变量生物活性毫无影响。

再次，问题研究的目的是进行变量选择，从而需要采取多个降维方法才能更好的处理超高维度变量。具体的是挑选出对生物活性具有显著影响的变量，需要多次采用相关性分析以及关联度分析才能达到挑选变量的目的，因此考虑二级变量降维方法进行降维来达到挑选最优特征变量的目的。

最后，构建第一级变量降维方法—皮尔森相关系数，在剩余数据集中，根据皮尔森相关系数过滤掉与生物活性相关性较低（相关系数小于 0.4）的变量实现第一级变量降维，可以得到相关性较强的 24 个分子描述符（变量）；再构建第二级降维方法—灰色关联分析，利用挑选出的 24 个分子描述符来对因变量生物活性进行关联度分析，求出关联度系数并筛选掉关联度系数小于 0.95 的变量，能够得到高度显著影响生物活性的 20 个变量，最后根据查阅文献综合确定了前 20 个对生物活性最具有显著影响的分子描述符（即变量）并给出变量对生物活性影响的重要排序表。

问题一的思路流程图如 3-1 图所示：

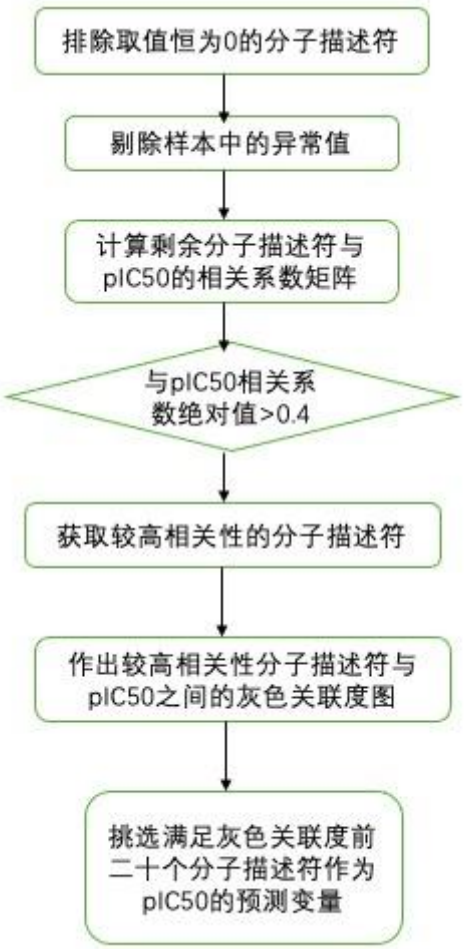

图 3-1 问题一思路流程图

3.2 原始数据说明

题目提供了 1974 个化合物以及对应的 729 个分子描述符，并给出了代表生物活性的二个指标分别为 IC_{50} 值和对应的 pIC_{50} 值。其中，1974 个代表化合物代表数据的条数，对应的 729 个分子描述符代表着影响化合物活性的自变量，C_{50} 值和对应的 pIC_{50} 值代表着生物活性即为因变量。IC_{50} 值越小代表生物活性越大，对抑制 ERα活性越有效，pIC_{50} 值越大表明生物活性越高。IC_{50} 值和对应的 pIC_{50} 值之间有一个换算公式为：

$$pIC_{50} = -\log_{10}\left(IC_{50} \times 10^{-9}\right) \tag{3.1}$$

其中，IC_{50} 的单位是 nM

3.3 选取代表生物活性指标

选取代表生物活性至关重要，直接影响挑选对生物活性影响的变量以及前20个对生物活性最具有显著影响的分子描述符（即变量）。根据所给的代表生物活性指标 IC_{50} 、pIC_{50} 以及对应的指标值，并且 IC_{50} 、pIC_{50} 之间具有换算关系，相当于 IC_{50} 、pIC_{50} 指标具有等价性。

观察所给的 IC_{50} 、pIC_{50} 指标值，发现 IC_{50} 值波动值非常大而 pIC_{50} 值波动值比较小，为了使后期模型更具鲁棒性以及更好的训练模型，从而采取以 pIC_{50} 特征作为代表生物活性指标。

3.4 滤除无关变量

无关变量是指那些不是某实验所需研究的，自变量与因变量之外的一切变量的统称，也称为非实验因子或无关因子。无关变量又称为控制变量，即实验中除了自变量以外，其它的可能引起实验结果改变的因素。

问题一中的1974个化合物的729个分子描述符为自变量，生物活性为因变量，其中选取 pIC_{50} 特征作为代表生物活性指标。自变量高达729个，数据的维度太高、数量太多，难免出现一些无关变量，从而采取一些数据预处理方法来删除掉无关变量具有重要意义。于是，采取了一些预处理的方法，利用编程工具成功删除掉无关变量多达225个。最后，将剩余的1974个化合物的504个分子描述作为后面模型的训练数据集。

3.5 基于皮尔逊相关分析的第一级变量降维模型

3.5.1 皮尔逊相关分析理论介绍

皮尔逊相关系数，又称为皮尔逊积矩相关系数，常用于度量二个变量 X 和 Y 之间的相关性，其中取值在-1 与 1 之间。一般用于分析二个连续变量之间的相关系数关系，其中公式表示为：

$$r=\frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2\sum_{i=1}^{n}(y_i-\bar{y})^2}} \tag{3.2}$$

其中，r 代表变量 X 和 Y 之间的皮尔逊相关系数，$x_i(i=1,2,\cdots,n)$ 代表变量 X 的元素，$y_i(i=1,2,\cdots,n)$ 代表变量 Y 的元素。

Pearson 相关系数的取值范围为 $-1 \le r \le 1$，根据计算的相关系数大小，判别因变量与自变量相关性大小标准如下表：

表 3-1 Pearson 相关系数的线性相关性判别准则

$\|r\|$	相关程度
$0.8 < \|r\| \le 1$	极强相关
$0.6 < \|r\| \le 0.8$	强相关
$0.4 < \|r\| \le 0.6$	中等程度相关
$0.2 < \|r\| \le 0.4$	弱相关
$0 < \|r\| \le 0.2$	极弱相关

3.5.2 皮尔逊相关分析模型建立

数据准备，首先选取前面剔除无关变量以及异常数据后的数据集作为皮尔逊相关模型的样本数据。

题目要求，找出变量对生物活性影响的重要性排序。采取了皮尔逊相关分析方法分别计算出剩余的 504 个分子描述与代表生物活性变量 pIC_{50} 之间的皮尔逊相关系数大小，依据相关系数大小对变量对生物活性影响的重要性进行排序。

变量对生物活性影响的重要性确定。皮尔逊相关系数 $|r| \le 0.4$ 表示变量之间的相关性较低，从而相互影响程度较弱。进而，计算出剩余的 504 个分子描述与代表生物活性变量 pIC_{50} 之间的皮尔逊相关系数大小，筛选掉皮尔逊相关系数 $|r| \le 0.4$ 分子描述符变量，得到与生物活性相关性较强的 24 个分子描述符，表明该 24 个分子描述符对生物活性具有重要影响。

结果统计，得到 504 个分子描述与代表生物活性变量 pIC_{50} 之间的皮尔逊相关系数大小，得到 24 个对生物活性具有重要影响的分子描述符。

3.5.3 第一级变量降维结果分析

根据皮尔逊相关模型所到的结果，可知 504 个分子描述与代表生物活性变量 pIC50 之间的皮尔逊相关系数大小[1]，**并在附录中给出有效变量对生物活性影响的重要性进行排序表**。分析结果，依据皮尔逊相关系数大小，得到 24 个对生物活性具有重要影响的分子描述符分别为 MDEC-23、MLogP、LipoaffinityIndex、maxsOH、minsOH、nC、nT6Ring、n6Ring、minsssN、BCUTp-1h、C2SP2、hmin、AMR、AMR、SwHBa、maxsssN、MDEC-22、SP-5、SaaCH、CrippenLogP、maxHsOH、C1SP2、nHaaCH、naaCH、ATSp4，并给出这 24 个分子描述符（变量）对生物活性影响的重要性进行排序表，如下表：

表 3-2 24 个分子描述符对生物活性影响的重要性排序

序号	分子描述符	皮尔逊相关系数大小
1	MDEC-23	0.538
2	MLogP	0.529
3	LipoaffinityIndex	0.492
4	maxsOH	0.467
5	minsOH	0.466
6	nC	0.460
7	nT6Ring	0.440
8	n6Ring	0.432
9	minsssN	0.431
10	BCUTp-1h	0.429
11	C2SP2	0.427
12	hmin	-0.426
13	AMR	0.425
14	SwHBa	0.423
15	maxsssN	0.421
16	MDEC-22	0.420
17	SP-5	0.419
18	SaaCH	0.419
19	CrippenLogP	0.412
20	maxHsOH	0.409
21	C1SP2	-0.407
22	nHaaCH	0.405
23	naaCH	0.405
24	ATSp4	0.401

计算出了所有分子描述与代表生物活性变量 pIC50 之间的皮尔逊相关系数大小，发现$|r|>0.4$ 分子描述符变量所占比例较少，说明筛选出变量对生物活性影响非常大。

为了更好的判断出题目所给分子描述是否对生物活性的影响性，下面给出了皮尔逊相关系数各部分的占比图，如下图（示例）：

图 3-2 Pearson 系数分布的饼图

3.6 基于灰色关联度分析的第二级变量降维模型

3.6.1 灰色关联度分析理论介绍

灰色关联度分析是一个对系统发展变化态势的定量描述和比较的方法，也是分析各个因素对结果的影响程度的方法，还是一种研究变量之间相似度以及相关关系的策略。关联度分析法，即根据因素间发展态势的相似或相异程度来衡量因素间关联的程度。

灰色关联度分析的具体步骤，首先，选取好参考数列与比较数列，其次，对参考数列与比较数列匠心无量纲化处理，然后，计算出变量与因变量之间的关联系数以及关联度，最后给出关联度排序。其中，参考数列也叫母序列，反映系统行为特征的数据序列，简而言之就是需要被比较的因变量；比较序列也叫子序列，影响系统行为的因素组成的数据序列，简而言之就是需要比较的自变量。关联系数表示的是比较数列与参考数列在某时刻关联程度的一种指标。关联度表示的是把各个类别的关联系数集中为一个均值。

3.6.2 灰色关联度分析模型建立

数据准备，首先选取皮尔逊相关模型所挑选出 24 个对生物活性具有重要影响的分子描述符，作为灰色关联度分析模型的样本数据。

题目要求，前 20 个对生物活性最具有显著影响的分子描述符（即变量）。利用了皮尔逊模型所挑出的对生物活性具有重要影响的变量，近一步更好的去挑选出前 20 个对生物活性最具有显著影响的变量。

选取代表生物活性变量 pIC_{50} 作为参考列、24 个对生物活性具有重要影响的分子描述符作为比较序列，用关联系数以及关联度来刻画分子描述符对生物活性

的影响的显著性，最后更好的表明显著影响，筛选掉关联度小于 0.91 的分子描述符，得到20个对生物活性最具有显著影响的分子描述。对应生物活性变量 pIC_{50} 与 24 个对生物活性具有重要影响的分子描述符之间的关联系数、关联度计算公式如下所示：

$$\varsigma_{i(k)}=\frac{\min\limits_{i}\min\limits_{k}|Z_{0(k)}-Z_{i(k)}|+\rho\min\limits_{i}\min\limits_{k}|Z_{0(k)}-Z_{i(k)}|}{|Z_{0(k)}-Z_{i(k)}|+\rho\max\limits_{i}\max\limits_{k}|Z_{0(k)}-Z_{i(k)}|} \tag{3.3}$$

$$\lambda_i=\frac{1}{n}\sum_{k=1}^{n}\varsigma_{i(k)},k=1,2,..24 \tag{3.4}$$

其中，$\varsigma_{i(k)}$ 代表生物活性变量 pIC_{50} 与第 i 个对生物活性具有重要影响的分子描述符之间的关联系数、λ_i 代表生物活性变量 pIC_5 与第 i 个对生物活性具有重要影响的分子描述符之间的关联系数、Z_0 代表参考列代表生物活性变量 pIC_{50} 、Z_i 代表 24 个对生物活性具有重要影响的分子描述符，$\min\limits_{i}\min\limits_{k}|Z_{0(k)}-Z_{i(k)}|$ 、$\max\limits_{i}\max\limits_{k}|Z_{0(k)}-Z_{i(k)}|$ 分别代表二级最小差和二级最大差

前 20 个对生物活性最具有显著影响的分子描述符确定。利用计算的 24 个对生物活性具有重要影响的分子描述符与代表生物活性变量 pIC_{50} 之间的关联度。更好的表明显著影响，筛选掉关联度小于 0.91 的分子描述符，得到 20 个对生物活性最具有显著影响的分子描述。

结果统计，得到 24 个对生物活性具有重要影响的分子描述符与代表生物活性变量 pIC_{50} 之间的关联系数以及关联度，筛选得到前 20 个对生物活性最具有显著影响的分子描述符。

3.6.3 第二级变量降维结果分析

根据灰色关联分析模型所到的结果，得到 24 个对生物活性具有重要影响的分子描述符与代表生物活性变量 pIC_{50} 之间的关联度，筛选得到前 20 个对生物活性最具有显著影响的分子描述符。为了更加直观判断 24 个对生物活性具有重要影响的分子描述符对代表生物活性变量 pIC_{50} 影响程度，给出了相应的灰色关联度取值排序表，如下 3-3 表：

表 3-3 24 个分子描述符与生物活性变量灰色关联度取值排序

序号	分子描述符	灰色关联度大小
1	BCUTp-1h	0.9854
2	MLogP	0.9853
3	AMR	0.9816
4	nC	0.9812
5	SP-5	0.9808
6	CrippenLogP	0.9804
7	MDEC-23	0.9782
8	ATSp4	0.9774
9	C2SP2	0.9768
10	n6Ring	0.9768
11	nT6Ring	0.9764
12	LipoaffinityIndex	0.9758
13	SwHBa	0.9727
14	nHaaCH	0.9723
15	naaCH	0.9723
16	SaaCH	0.9697
17	maxsOH	0.9674
18	minsOH	0.9672
19	maxHsOH	0.9628
20	MDEC-22	0.9589
21	C1SP2	0.9145
22	minsssN	0.9126
23	maxsssN	0.9125
24	hmin	0.8721

分析表格得出，前 20 个分子描述符与代表生物活性变量 pIC_{50} 之间的关联度均大于 0.95，接近于 1，从而判断出前 20 个分子描述符对生物活性具有显著影响。

3.7 确定前 20 对生物活性最具有显著影响的分子描述符

选取与代表生物活性变量 pIC_{50} 之间关联度大于 0.95 的变量，作为前 20 个对生物活性最具有显著影响的分子描述符。前 20 个对生物活性最具有显著影响的分子描述符分别为 **BCUTp-1h、MLogP、AMR、nC、SP-5、CrippenLogP、MDEC-23、ATSp4、C2SP2、n6Ring、nT6Ring、LipoaffinityIndex、SwHBa、nHaaCH、SaaCH、maxsOH、minsOH、maxHsOH、MDEC-22。**

3.8 基于皮尔逊相关分析、灰色关联度分析的二级变量降维模型评价

根据灰色关联分析模型，发现计算出 24 个对生物活性具有重要影响的分子描述符与代表生物活性变量 pIC_{50} 之间的关联度都不低于 0.87，侧面评价出皮尔逊相关分析模型筛选出的变量，具有对生物活性影响较大的性质，从而得出皮尔逊相关分析模型较好。

依据灰色关联分析模型所算出的关联度大小，并选择与生物活性关联度大于 0.95 的分子描述符，作为最具有显著影响的分子描述符。根据查阅的文献知识，可知挑选的变量合理，才得到关联度接近于 1 的结果，表明灰色关联分析模型较好。

4 问题二的分析与求解

4.1 问题分析

针对问题二，题目要求选择不超过 20 个分子描述符变量，构建出化合物对 ERα生物活性的定量预测模型，以及利用建立的定量模型对给出的 50 个化合物进行 IC_{50} 值和对应的 pIC_{50} 值预测并填入对于表中。

首先，选取第问题一中挑选的前 20 个对生物活性最具有显著影响的分子描述符（即变量）来作为预测模型的训练数据。并在训练之前，需要判断出描述符变量与代表 ERα生物活性的指标 pIC_{50} 之间具有是否为线性关系以及对应数据是否标准化。

进而，采用散点图分析出描述符变量与代表 ERα生物活性的指标 pIC_{50} 之间具有高度的非线性关系。判断出常规的标准化处理公式处理数据效果不好，最终，选择拉依达准则（3σ 法则）来进行数据的标准化。

再次，利用所标准化的数据以及对应数据之间高度非线性特性，采取多元线性回归、双隐层 BP 神经网络、随机森林算法，构建预测代表 ERα生物活性指标 pIC_{50} 值的定量预测模型。经过对三种模型效果进行分析得到：多元线性回归预测模型的均方误差 MSE 为 1.584，平均绝对百分比误差 MAPE 为 35.69% ；双隐层 BP 神经网络预测模型的均方误差 MSE 为 0.929，平均绝对百分比误差 MAPE 为 17.46% ；随机森林预测模型的均方误差为 MSE 为 0.287，平均绝对百分比误差 MAPE 为 3.71% 。因此，根据横向对比三种预测模型的预测的准确性以及鲁棒性，确定化合物对 ERα生物活性的随机森林定量预测模型。

最后，利用建立的随机森林定量预测模型来对对文件“ERα_activity.xlsx”的 test 表中的 50 个化合物进行对应的 pIC_{50} 值预测，并利用 EXCEL 表工具和 IC_{50} 值与 pIC_{50} 值关系转换公式，计算出 IC_{50} 值。最终将预测的 pIC_{50} 值以及 IC_{50} 值分别填入 pIC50 列、IC50_nM 列。

问题二的思路流程图如 4-1 图所示：

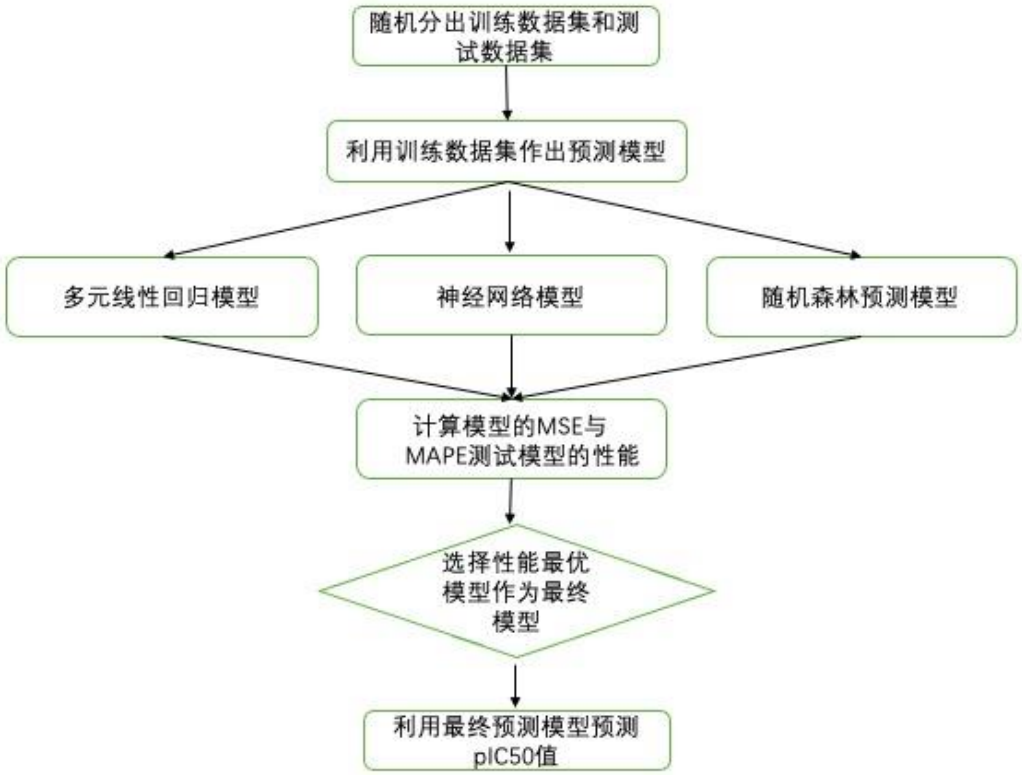

图 4-1 问题二思路流程图

4.2 分析数据

4.2.1 选取训练数据

对于预测模型，训练数据非常重要，从而选取第问题一中挑选的前 20 个对生物活性最具有显著影响的分子描述符（即变量）来作为预测模型的训练数据。

4.2.2 分析训练数据间的线性关系

本文分析了选取变量与生物活性之间的线性关系，在此基础上选择合理的预测模型。随机抽取 1974 个化合物的分子描述符信息（见附录），将 pIC50 值与随机抽取的变量作散点图，初步观察选取变量与生物活性的线性关系。如下图所示：

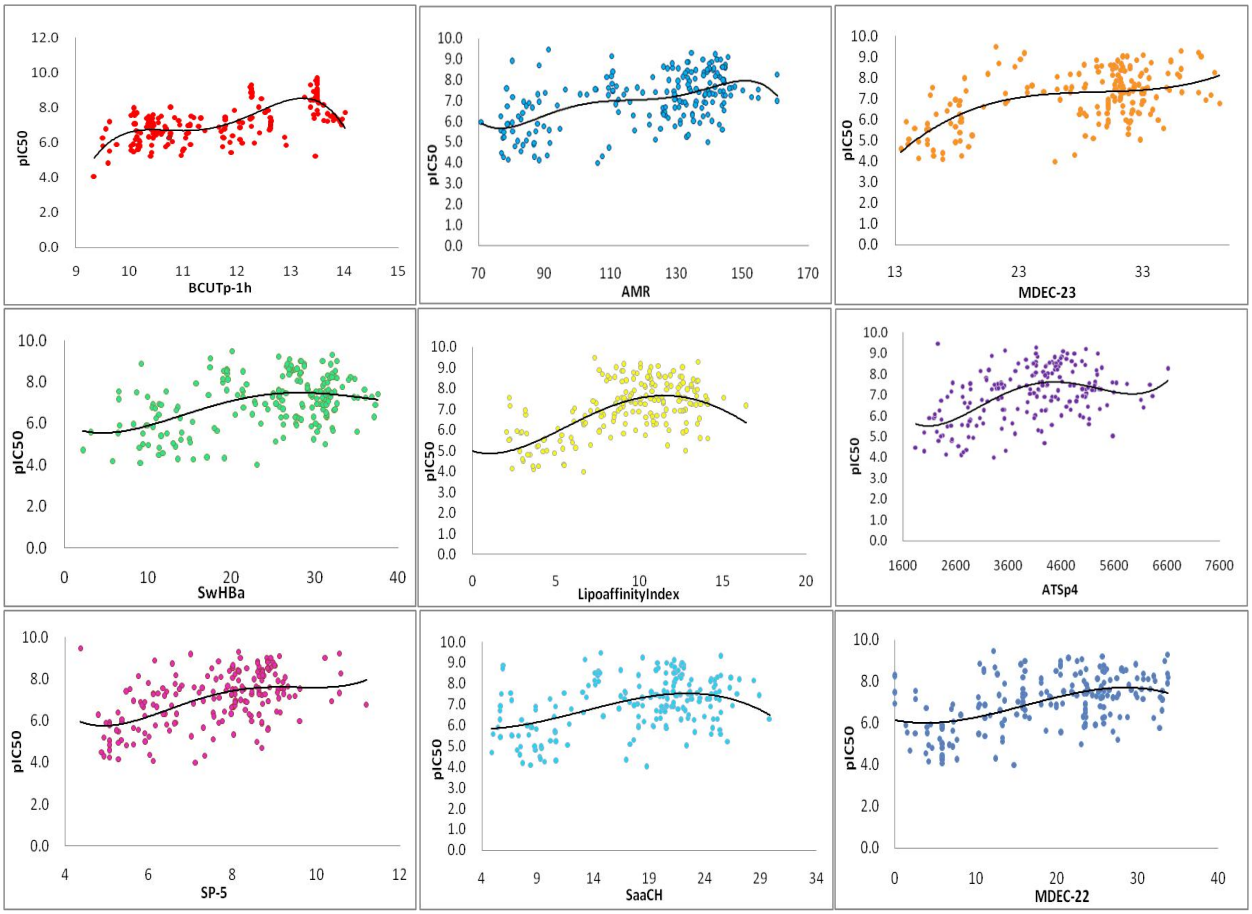

图 4-2　部分主要变量与生物活性之间的散点图

根据 4-2 图中点的排列可以初步判断出选取的变量和生物活性之间具有高度的非线性关系。

4.2.3 标准化训练数据

（1）常规标准化处理

数据之间的量纲差异较大，为了提高模型预测的准确性，需要根据式（4.1）对数据做标准化处理。

$$\hat{x} = \frac{x - x_{mean}}{x_{std}} \tag{4.1}$$

其中，x 为自变量，x_{mean} 为样本均值，x_{std} 为自变量的样本标准差。

（2）拉依达准则(3σ 法则)标准化处理

一组数据存在误差，误差在一定范围内可被认为是随机误差，一旦超过某个特定范围则被认为是粗大误差。拉依达准则通过对数据处理计算得到标准偏差，剔除超过区间误差的数据，可以减少样本中的异常数据，提高模型预测的精度。运用样本标准差进行模型的计算存在不足，在实际应用中，其钢性（稳定性）不强，因此采用 3σ 法则对标准化后的数据进一步处理。

3σ 法则如下表：

表 4-1 3σ 法则对照表

范围	概率
$(\mu-\sigma,\mu+\sigma)$	0.6826
$(\mu-2\sigma,\mu+2\sigma)$	0.9544
$(\mu-3\sigma,\mu+3\sigma)$	0.9974

根据表 4-1，若一个随机变量的观察值几乎都落在区间 $(\mu-3\sigma,\mu+3\sigma)$ 中，就可以得到

$$\hat{\sigma}=\frac{\max(x)-\min(x)}{6}\quad . \tag{4.2}$$

在式（4.2）中，$\hat{\sigma}$ 为自变量标准差的估计值，$\max(x)$ 为自变量观察值中的最大值，$\min(x)$ 为自变量观察值中的最小值。根据拉依达准则处理后得到的数据，具有更好的稳定性。

4.3 多元线性回归模型的建立与求解

多元线性回归模型可以用来研究多个自变量对一个因变量的影响大小，如果两者的关系可以用线性形式来刻画，则该模型可以取得较好的预测效果。本文挑选了相关性系数大于 0.4 的分子描述符作为自变量，化合物的生物活性与自变量之间存在中等程度以上的线性关系，因此初步建立多元线性回归模型对数据进行预测，便于与其它方法作比较说明。

4.3.1 多元线性回归模型的建立

多元线性回归模型用于描述自变量和因变量之间的随机线性关系，具体计算公式为

$$f(x)=w_1x_1+w_2x_2...+w_dx_d+b\,, \tag{4.3}$$

在式（4.3）中，$x_1,x_2,...,x_d$ 是自变量，$f(x)$ 是因变量，w_1，$w_2,...w_d$ 是回归系数，b 是随机误差项。若用偏导的方式来求解方程过程较为繁琐，进而考虑将其转化为向量之间的运算，直接对向量求导可以得到正则方程。

假设在多元线性回归模型中有 m 个样本，那么任意的样本 $x=(x_1,x_2,...,x_d)^T$ 就有 d 个维度，对应的回归系数的向量表达式为 $w=(w_1,w_2,...,w_d)^T$，进一步可以得到表达式

$$y_i = w^T x_i + b. \tag{4.4}$$

接着将随机误差项考虑到计算过程中，令 $\hat{w} = (w,b)^T$ 并把所有的自变量合为一个矩阵 X 。令因变量 $y = (y_1, y_2, ..., y_m)^T$ ，则有

$$E(\hat{w}) = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2 = (y - X\hat{w})^T (y - X\hat{w}) \tag{4.5}$$

根据式（4.5）,可以经一步得到

$$\hat{w}^* = \arg\min (y - X\hat{w})^T (y - X\hat{w}). \tag{4.6}$$

4.3.1 多元线性回归模型的求解

模型中的样本量较大，为了提高模型的运算速率，将因变量 pIC_{50} 转化为列向量，自变量转化为一个矩阵，采用矩阵进行运算。根据式（4.6），本文先求解参数向量 $\hat{w}^*$,然后通过得到的模型计算出 MSE 值为 1.584，MAPE 值为 3.569%。根据模型的评价指标，可以知道多元线性回归求解效果非常不好。

4.4 双隐层 BP 神经网络定量预测优化模型的建立与求解

根据多元线性回归模型的误差大小，可以看出模型的预测效果并不理想。选取的 20 个变量与生物活性之间虽然存在一定的线性关系，但该相关性较弱。因此，本文尝试构建非线性的神经网络模型对数据进行预测。

4.4.1 BP 神经网络简介

人体的神经元有轴突、树突和神经核，数学家们受到人的神经元原理的启发构建了神经网络模型。其中，BP 神经网络，也称为误差反向传播神经网络,是神经网络模型中较为广泛运用的一种模型，本文基于该方法建立双隐层 BP 神经网络定量预测模型。[2]

（1）BP 神经网络理论

BP 神经网络一般由三部分构成，即输入层、隐含层和输出层。BP 神经网络就是通过误差的反向传播来不断地校正神经网络的权值和阈值，使得模型的输出误差达到最小。当输入一个样本并产生输出时，会根据所有网络最小化输出的均方差来训练网络。其中，各个神经元单位输出的误差平方和称为均方误差，计算公式为

$$E(w)=\frac{1}{2}\sum_{k=1}^{L}(y(k)-y_d(k))^2. \tag{4.7}$$

在式（4.7）中，L 为训练集中向量的个数，$y_d(k)$ 为期望的样本输出，$y(k)$ 是神经网络对于输出的第 k 个值，w 是权值向量。

接着，由梯度下降法可以得到

$$\Delta w=\nabla E(w)|_{w=w(t)}. \tag{4.8}$$

将输入层和隐含层的权值设为 $w_1(t)$，则可以得到输入层和隐含层的权值调整公式为

$$\Delta w_1(t)=-\eta\frac{\partial E}{\partial w_1(t)}. \tag{4.9}$$

将隐含层和输出层的权值设为 $w_2(t)$，则可以得到隐含层和输出层的权值调整公式为

$$\Delta w_2(t)=-\eta\frac{\partial E}{\partial w_2(t)}. \tag{4.10}$$

（2）BP 神经网络的具体计算步骤

在 BP 神经网络中，反向传播实际上就是通过不断的对损失函数关于对应的参数求导。该方法利用梯度下降法，寻找“最优下山”方向，通过设置的步长，对随机生成的参数进行优化，以此不断地优化模型。在该模型中，优化准则就是使得 MSE 逐渐减小。

模型建立后会随机生成参数，利用反向传播来迭代优化参数，使得损失函数最小化。反向传播的基本思想就是通过计算输出层与期望值之间的误差来调整网络参数，从而使得误差变小。

据此，我们可以得到 BP 神经网络的具体计算步骤如下：

Step1：BP 神经网络的初始化，设置各个权值和阈值的初始值并确定各层的节点个数。

Step2：对每一个样本进行学习和训练，即重复步骤 3 到步骤 5。

Stsp3：根据输入的样本来计算隐含层和输出层的神经元输出。

Step4：根据期望输出值和实际输出值的差值大小，来计算输出层和隐含层的误差大小。

Step5：根据步骤 4 中计算出的误差值，来更新输入层和隐含层以及隐含层和输出层之间的权值大小。

Step6：判断误差函数是否收敛到所给定的学习精度内，若满足精度大小则学习结束，若不满足则返回步骤 2 重新训练样本。

（3）激活函数

神经网络模型模仿了人体脑神经的工作过程，对于所收集的 n 个信息，就会有 n 个对应的参数与 n 个信息进行运算求和。根据其求和结果，神经元会在非线性的激活函数中进行判别，然后把输出的结果传递给下一个神经元。因此，选取一个合适的激活函数对于神经网络模型是十分重要的。

在建立神经网络的过程，常用到的激活函数有 Sigmoid 非线性激活函数

$$\sigma(x)=\frac{1}{(1+e^{-x})}, \tag{4.11}$$

Sigmoid 非线性激活函数将输入的实数值压缩到$[0,1]$范围内，特别地，绝对值大的负数被映射成 0，绝对值大的正数被映射成 1.

此外，神经网络中也常用 Tanh 函数作为激活函数，它是 Sigmoid 非线性激活函数一种变形，其形式为

$$\tanh(x)=2\sigma(2x)-1. \tag{4.12}$$

该激活函数与Sigmoid非线性激活函数的差别在于，它将输入值压缩到$[-1,1]$的范围内。

4.4.2 双隐层 BP 神经网络定量预测模型的建立

本文建立了双隐层 BP 神经网络，用于预测附件中 50 个化合物样本的生物活性。

（1）数据整理

本文根据附件 2 挑选了相关性显著 20 个分子描述符，此步需要对 1974 组数据进行处理，我们选取了 80%和 20%的比例将数据划分为训练集和测试集。

（2）网络结构层的设计

1）**输入输出层**：输入之前步骤中选取的 20 个主要变量，输出化合物的生物活性。因此，输入的神经元个数为 20，输出的神经元个数为 1.

2）**隐含层**：在建立神经网络模型中，确定隐含层的个数和每个隐含层神经元的个数都十分重要。如果隐含层或神经元个数选取过多，会造成计算量较大和出现过拟合的问题；如果隐含层或神经元个数选取过少，则会造成预测精度不好的后果。隐含层和各层神经元个数的确定都有实际问题的复杂程度有关，目前还没有具体的公式可以计算，所以本文通过反复调试模型来确定出隐含层的个数和各层神经元的个数。本文确定出隐含层的数量为 2，各隐含层的神经元个数分别为 9 和 3.因此，本文建立的双隐层 BP 神经网络定量预测模型共需要的参数个数为 223 个。

3）**激活函数的选取**：以上介绍的两种激活函数均存在梯度饱和的缺点，因此本文建立的双隐层 BP 神经网络定量预测模型采用了 relu 函数作为激活函数，该函数的表达式为

$$f(x)=\max(0,x), \tag{4.13}$$

若输入值为负数则输出值全为 0，若输入值为正数则输入值和输出值相等。该方法抑制了同一时间内部分神经元的激活，从而使得网络变得稀疏，提高了计算的效率。

4）**损失函数**：本文选取 MSE 作为损失函数，MSE 为凸函数有优良的性质，其局部最优解就是全局最优解。

4.4.3 双隐层 BP 神经网络定量预测模型的优化

常用的剪枝算法有惩罚函数法、灵敏度计算法、相关性剪枝算法等，其中惩罚函数法是使用最普遍的剪枝算法之一[3] 。

（1）引入惩罚项

为了防止神经网络训练模型过拟合，本文对双隐层 BP 神经网络定量预测模型的损失函数添加惩罚项：

$$L = MSE + \lambda * \|W\|, \tag{4.14}$$

在式(4.14)中，$\|W\|$ 表示参数的 2-范数，λ 为惩罚系数。

（2）惩罚系数的确定

本文通过不断调试参数，根据输出结果来大致确定出惩罚系数的取值。

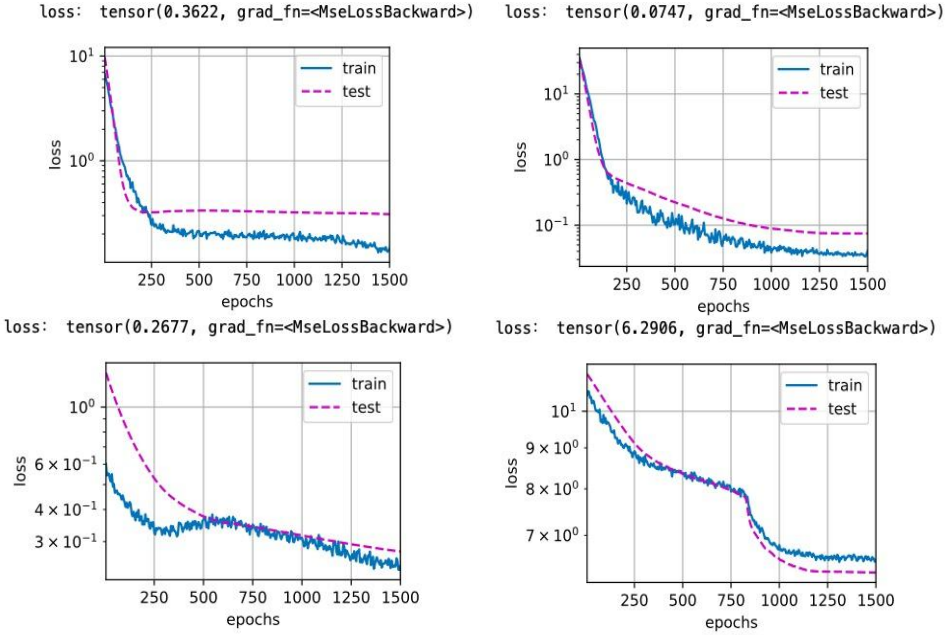

图 4-3 不同参数下测试集与训练集数据的曲线对比

以上 4 个图从左到右、从上到下依次为参数为 0,1,2,3 时，测试集数据与训练集数据的曲线。观察图 4-3，可知选取参数为 1 时，测试集与训练集曲线走势大致相同，因此本文选取惩罚系数为 1。

4.4.3 双隐层 BP 神经网络定量预测优化模型的计算

本文运用 Python 编写程序对双隐层 BP 神经网络定量预测优化模型进行求解。

（1）模型实现

采用四层结构的 BP 神经网络优化模型，设置输入神经元个数等于输入分子描述符个数为 20；输出神经元个数等于输出结果化合物生物活性为 1；选取 relu 函数作为激活函数；选取 MSE 作为损失函数；设置最大迭代次数为 1000 次，学习速率为 0.001。

（2）模型求解结果

本文选取的四层结构 BP 神经网络优化模型经过重复学习后达到期望误差，就可以结束学习。将显著性相关变量输入网络就可以得到化合物生物活性的预测结果见图 4-4。

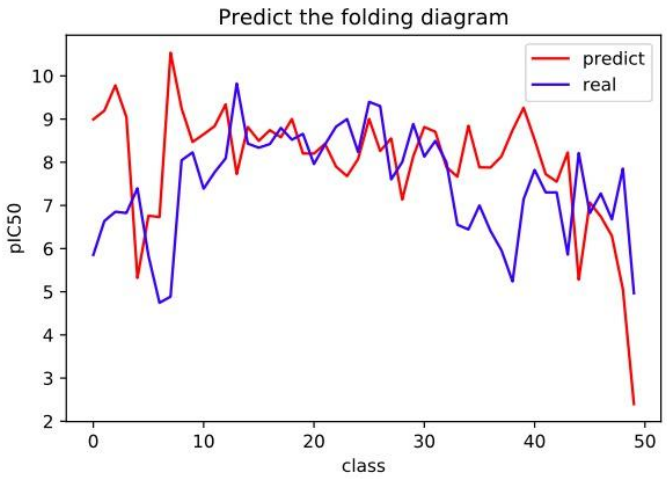

图 4-4 基于 BP 神经网络定量预测 pIC_{50} 结果与真实值对比

图 4-4 为基于双隐层 BP 神经网络定量预测优化模型的 pIC_{50} 值的预测结果，并得到该模型的 MSE 值为 0.929,MAPE 值为 17.46%。

4.5 基于随机森林定量预测模型

4.5.1 随机森林定量预测理论介绍

随机森林算法是一种比较新的机器学习的学习算法，对于预测是采用决策树分类做法，也是一种特殊的自主抽样集成，袋装法。

神经网络容易过拟合，而随机森林利用随机抽样和随机抽取特征的特性，可以非常好的防止出现过拟合现象，以及可以克服变量维度灾难。并且，随机森林预测模型相对简单，从而采取该算法进行定量预测，并分析预测效果。

随机森林建立过程为：最开始的训练集合中有 N 个样本，并且样本对应的

维度为 K 维。然后，从数据集中有放回的随机进行 m 次采样，根据所采样的样本可以生成 m 个训练子集。再次，每个训练的自己都可以生成一棵决策树，即共生成 m 个分类树形成森林。同时，对于每个决策分类树，从树的每个节点处的 M 个特征中，随机选择 s 个特征，再按照一定的结点不纯度最小原则一直进行完全分裂，直至该节点的所有的训练的样本均属于同样的一类。最后依据生成的多个决策分类器，对相关变量进行预测。

4.5.2 随机森林定量预测模型建立

数据准备，随机森林算法采用的是决策树分类做法，无需对数据进行标准化处理。选取第问题一中挑选的前 20 个对生物活性最具有显著影响的分子描述符（即变量）来作为预测模型的训练数据。

题目要求，选择不超过 20 个分子描述符变量，构建化合物对 ERα生物活性的定量预测模型。故采用随机森林算法对对 ERα生物活性的定量预测，并判断出预测效果。

随机森林建模过程，可知训练数据集中有 N 个样本，且每个样本中有 20 个对生物活性最具有显著影响的分子描述符（即 24 维特征）。从 N 个样本中有放回随机的抽取 m 次，一共抽取 t 个样本，从而可以生成 m 个训练的子集。而每个子集都可以形成一个决策树，从而生成 m 棵决策树并形成森林。再在单个决策树中的每个节点处挑选 s 个分子描述符（特征），一直分裂，直至训练数据样本归属为一类。最终，根据生成的多个决策分类器进行预测。

结果统计，可以预测出 50 个化合物进行 IC_{50} 值和对应的 pIC_{50} 值，并得到预测的均方误差 MSE、平均绝对百分比误差 MAPE。

4.5.3 随机森林定量预测结果分析

利用随机森林定量预测模型，预测结果如下 4-5 图所示：

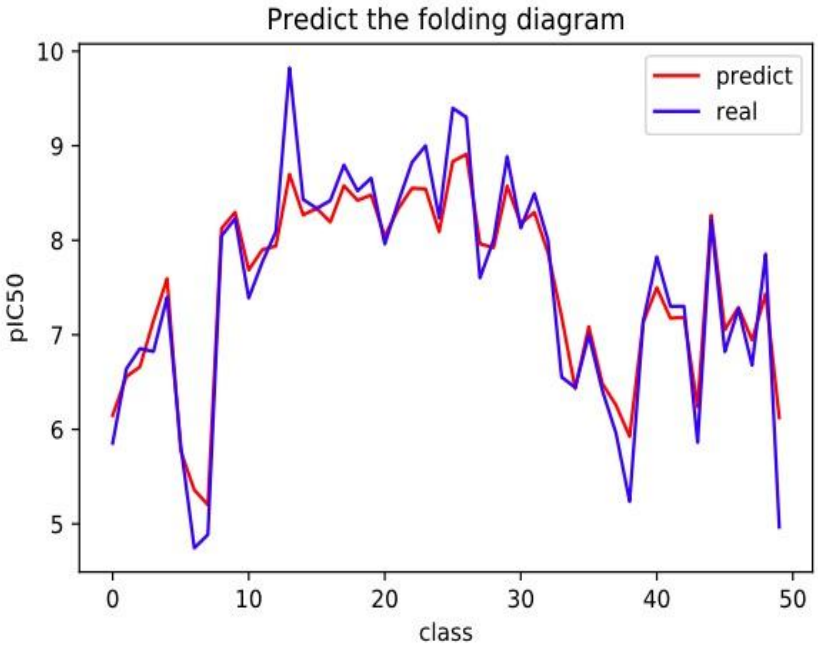

图 4-5 基于随机森林定量预测 pIC_{50} 结果与真实值对比

随机森林定量预测预测模型的相对百分比误差分析图如下 4-6 图为：

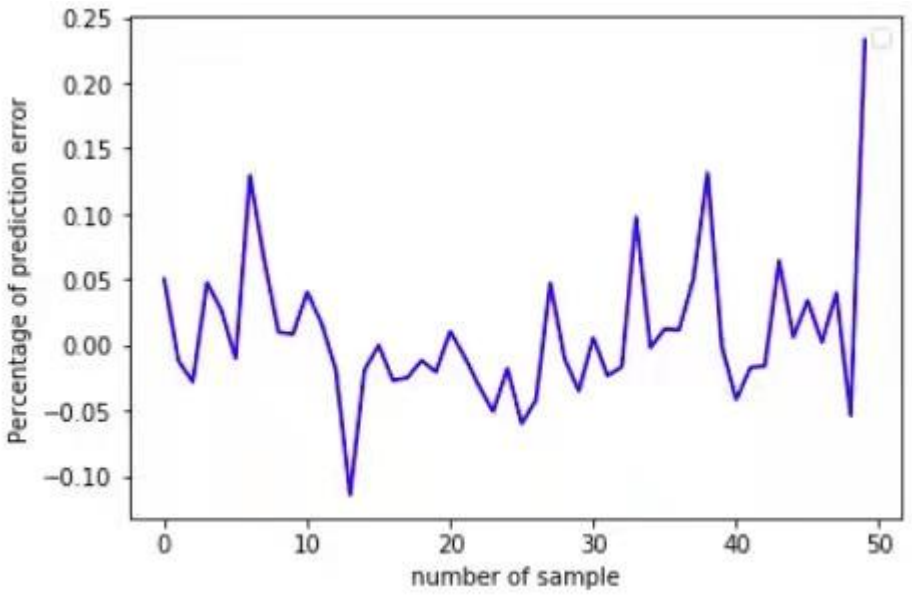

图 4-6 各样本的相对百分比误差分析

为了验证模型的预测效果，采取了随机抽样方法，从训练集中随机预留了 50 个数据作为验证模型好坏的测试集，其余的样本全部作为训练集。可得，相对百分比误差最高时所对应的样本为 50 号样本，为 0.24%，大多数样本集中在 0.10%以下，从而预测 pIC_5 值效果较好。随机森林预测模型的均方误差为 MSE 为 0.287，平均绝对百分比误差 MAPE 为 3.71%。模型的均方误差 MSE 0.287，表明模型的鲁棒性比较好；模型的平均绝对百分比误差 MAPE 低于 5%，说明模型预测的精度非常好。

4.6 模型评价指标

为了更好的表明模型的预测效果，本题选取了均方误差（MSE）、平均绝对百分误差（MAPE）来评价化合物对 ERα生物活性的定量预测模型好坏。

4.6.1 均方误差

均方误差是指参数估计值与参数真实值之差平方的期望值，反映的是估计量与被估计量之间差异程度的一种度量，取值范围为 $[0,+\infty)$ 。MSE 值越小，表明模型越好。

$$\mathrm{MSE}=\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i-y_i)^2 \qquad (4.15)$$

其中 $\hat{y}_i$ 表示预测值， y_i 表示真实值

4.6.2 平均绝对百分误差

平均绝对百分比误差是用于评估预测性能的最受欢迎的指标之一，本身就是用来衡量预测准确性的统计指标，取值范围为$[0,+\infty)$。MAPE 值越小，表明模型预测的准确度越高。

$$MAPE=\frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{\hat{y}_i-y_i}{y_i}\right| \tag{4.16}$$

其中$\hat{y}_i$表示预测值，y_i表示真实值

4.7 定量预测模型的评价以及最终预测模型的选取

为了更好的 ERα生物活性的预测效果，对比多元线性回归、双隐层 BP 神经网络、随机森林三种定量预测模型的预测效果，最终确定预测模型为随机森林定量预测模型。以下是三种模型关于 MSE、MAPE 模型评估表以及可视化的评价指标对比图。

表 4-2 定量预测模型评估表

定量预测模型	均方误差（MSE）	平均绝对百分误差（MAPE）
多元线性回归	1.584	35.69%
双隐层 BP 神经网络	0.929	17.46%
随机森林	0.287	3.71%

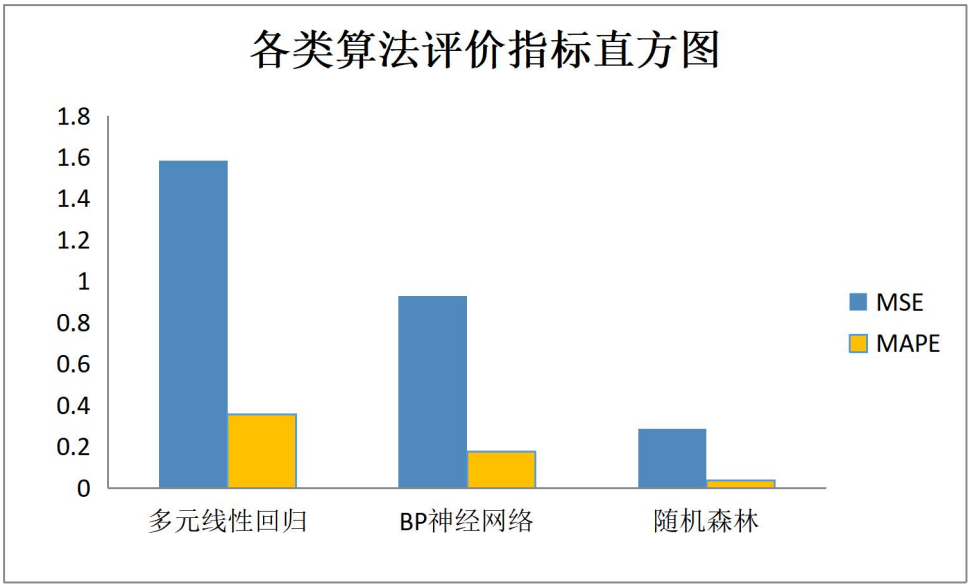

图 4-7 各类定量预测模型评价指标

根据对比结果，可知随机森林定量预测模型预测化合物对 ERα生物活性数据具有非常好的效果，平均绝对百分误差 MAPE 甚至小于 5%，表明预测准确性也非常高。因此，采用随机森林定量预测模型预测出 test 表中的 50 个化合物进行对应的 pIC_{50} 值，并利用公式求出 IC_{50} 值。

预测出 test 表中的 50 个化合物进行对应的 pIC_{50} 值，并利用公式求出 IC_{50} 值如下表 4.3：

表 4.3 test 表中的 50 个化合物进行 IC_{50} 值和对应的 pIC_{50} 值预测值

序号	IC50_nM	pIC50
1	29.2041	7.5346
2	18.5206	7.7323
3	17.5970	7.7546
4	19.1952	7.7168
5	19.4065	7.7121
6	13.8705	7.8579
7	21.4795	7.6680
8	29.0117	7.5374
9	14.2691	7.8456
10	9.1656	8.0378
11	12.6824	7.8968
12	13.7621	7.8613
13	17.5253	7.7563
14	16.6326	7.7790
15	21.0835	7.6761
16	20.0982	7.6968
17	30.0405	7.5223
18	39.4390	7.4041
19	51.5502	7.2878
20	2.0023	8.6985
21	19.0963	7.7191
22	15.4264	7.8117
23	61.8854	7.2084
24	30.5765	7.5146
25	62.1657	7.2064
26	21.9429	7.6587
27	20.3121	7.6922
28	13.0516	7.8843
29	108.7948	6.9634
30	36.4318	7.4385
31	2011.6503	5.6964
32	1873.3823	5.7274
33	2167.5655	5.6640
34	1642.0893	5.7846
35	1427.7143	5.8454

36	429.4350	6.3671
37	442.1978	6.3544
38	729.6528	6.1369
39	2751.0223	5.5605
40	2314.3146	5.6356
41	2252.7701	5.6473
42	2277.7894	5.6425
43	2360.3121	5.6270
44	2291.7964	5.6398
45	2252.7701	5.6473
46	29.5835	7.5290
47	15.5446	7.8084
48	17.6258	7.7539
49	34.0562	7.4678
50	35.5490	7.4492

5 问题三的分析与求解

5.1 问题分析

针对问题三，题目要求利用给出的 729 个分子描述符，针对 1974 个化合物的 ADMET 数据分别构建化合物的 Caco-2、CYP3A4、hERG、HOB、MN 的分类预测模型，并利用建立好的分类模型对 test 表中 50 个化合物分别预测出 Caco-2、CYP3A4、hERG、HOB、MN 值。

首先，依据提供的 729 个分子描述符以及 1974 个化合物的 ADMET 数据（Absorption 吸收、Distribution 分布、Metabolism 代谢、Excretion 排泄、Toxicity 毒性），判断出 Caco-2、CYP3A4、hERG、HOB、MN 为二值类别变量。从而，需要选择二分类方法先找出影响 Caco-2、CYP3A4、hERG、HOB、MN 的分子描述符，将其作为自变量，再利用二分类预测法分别计算出化合物中 Caco-2、CYP3A4、hERG、HOB、MN 值。

其次，基于 Caco-2、CYP3A4、hERG、HOB、MN 为二值类别变量，利用随机森林选择最优特征的方法，分别计算出影响 Caco-2、CYP3A4、hERG、HOB、MN 的特征重要性，并筛选出特征重要性大于 0.008 的分子描述符。然后，将选出的分子特征符作为自变量，分别用随机森林二分类、支持向量机二分类算法（SVM）对化合物的 Caco-2、CYP3A4、hERG、HOB、MN 的建立分类预测模型。

再次，筛选出影响 Caco-2、CYP3A4、hERG、HOB、MN 的代表性分子特征符。

最后，对随机森林二分类、支持向量机二分类模型预测效果进行分析得到：Caco-2 的支持向量机预测模型的准确率为 85.81% ，随机森林二分类预测模型的预测准确率为 90.91% ； CYP3A4 的支持向量机预测模型的准确率为 89.31% ，随机森林二分类预测模型的预测准确率为 90.68% ； hERG 的支持向量机预测模型的准确率为 89.80% ，随机森林二分类预测模型的预测准确率为 92.52% ；HOB 的支持向量机预测模型的准确率为 91.85% ，随机森林二分类预测模型的预测准确率为 81.11% ； MN 的支持向量机预测模型的准确率为 85.29% ，随机森林二分类预测模型的预测准确率为 91.83%。通过对比二种模型预测效果，对于 Caco-2、CYP3A4、hERG、MN 选取随机森林二分类预测模型，对于 HOB 选取支持向量机二分类预测模型，对文件“ADMET.xlsx”的 test 表中的 50 个化合物进行相应的预测，并将结果填入“ADMET.xlsx”的 test 表中对应的 Caco-2、CYP3A4、hERG、HOB、MN 列。

问题三的思路流程图如 5-1 图所示：

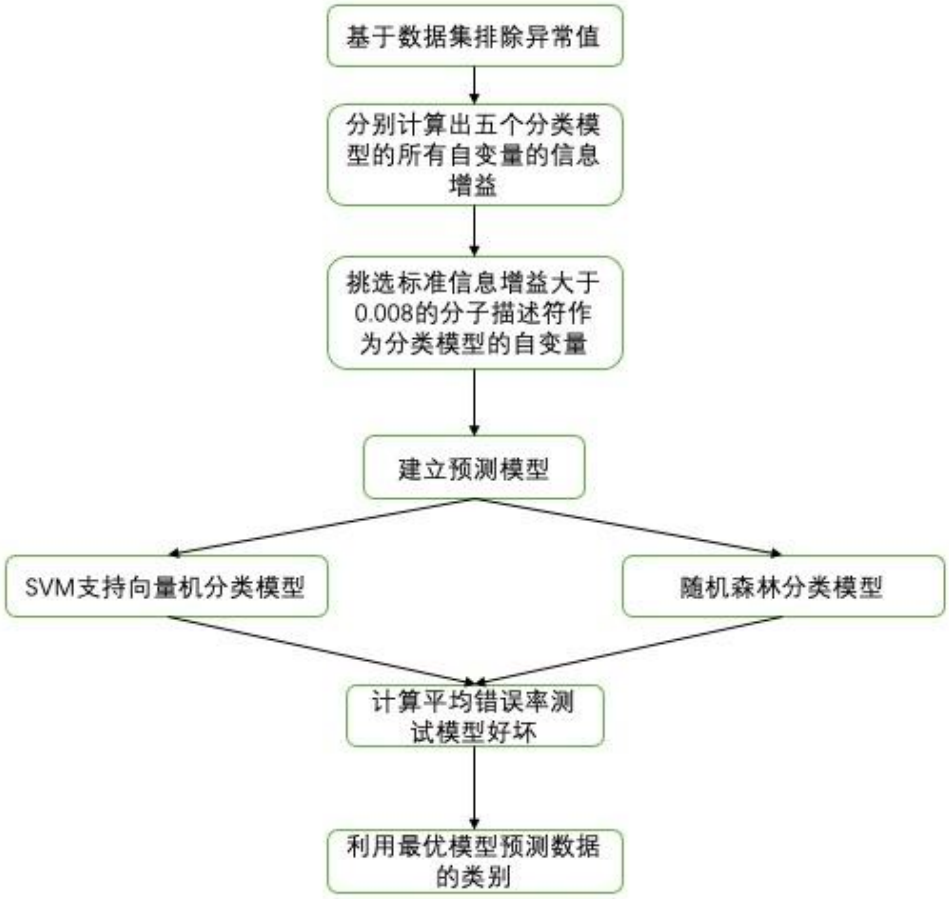

图 5-1 问题三思路流程图

5.2 分析数据

附件 4 中给出了 1974 个化合物的 ADMET 数据，为了更好建立模型需要将已知的 729 个变量维度降低，筛选出更加具有代表性数据有利于分类预测模型的训练。

经过分析可知，附件 4 所给数据的 974 个化合物的 Caco-2、CYP3A4、hERG、HOB、MN 值均为二分变量。

5.3 基于随机森林的分子描述符特征筛选模型

本题中所给的 1974 个化合物中的 729 个分子描述符具有高度非线性、数量多，共线性的特点，这与随机森林算法应用条件恰好相符，因此考虑建立基于随机森林的分子描述符特征筛选模型，分别挑选出影响 Caco-2、CYP3A4、hERG、HOB、MN 二分变量的具有显著性的分子描述符。

5.3.1 特征重要性计算

在随机森林的分子描述符特征筛选模型中，分子描述符的重要性其实就是每个分子描述符在随机森林的每棵树上贡献度的平均值，通过比较它们大小，来确定每个分子描述符的重要性。

袋外数据是指每次在构建决策树时没有被利用，而是被用于评估对决策树的性能。计算模型的预测错误率，这个错误率称为袋外数据误差：

$$OobErrorT_i^j = |e_1 - e_2|, \tag{5.1}$$

在式中，e_1为袋外数据样本得到的误差，e_2为随机打乱袋外数据得到的误差，$OobErrorT_i^j$为特征 j 对随机森林中树 i 的袋外数据误差。根据袋外数据

$$x_{j,importance} = \frac{\sum_{i=1}^{T} |e^j{}_{i,2} - e^j{}_{i,1}|}{T} \tag{5.2}$$

$x_{j,importance}$表示分子描述符 j 的特征重要性，T 表示随机森林中所有树的总数量，$|e^j{}_{i,2} - e^j{}_{i,1}|$表示分子描述符 j 对产生的随机森林中第 i 棵树的袋外数据误差，$e^j{}_2$表示对应的袋外数据样本的误差，$e^j{}_{i,1}$表示随机打乱袋外数据中第 j 个分子描述符得到的误差。

本文利用特征重要性来刻画分子描述符对 Caco-2、CYP3A4、hERG、HOB、MN 的影响程度，从而筛选出影响 Caco-2、CYP3A4、hERG、HOB、MN 的特征重要性指标。

5.3.2 随机森林的计算步骤

基于随机森林的分子描述符特征筛选模型包含建立随机森林包含如下步骤：

Step1：设定训练集为 N，根据 bootstrap 法有放回地随机抽取 k 个新的自助样本集，在此基础上生成 k 棵分类树，每次未被抽到的样本组成了 k 个袋外数据；

Step2：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 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 个相互独立的决策树；

Step3：生成的多棵树组成了随机森林，若干棵决策树是相互独立的，其重要性大小是一样的，不需要考虑决策树的权值。

根据以上所描述的随机森林计算步骤，我们可以得到随机森林算法的流程图如下图 5-2 所示：

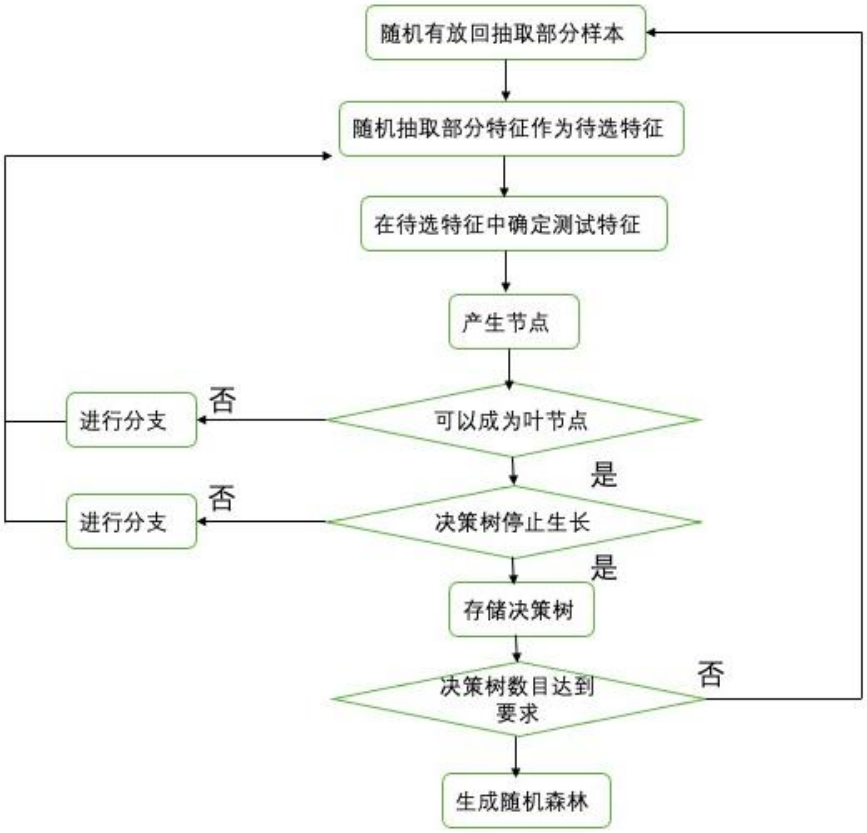

图 5-2 随机森林算法的流程图

5.3.3 筛选分子描述符变量

计算出特征重要性，再归一化求出相对特征重要性，由此挑选出影响 Caco-2、CYP3A4、hERG、HOB、MN 的分子描述符变量。利用随机森林选择最优特征的方法，分别计算出影响 Caco-2、CYP3A4、hERG、HOB、MN 的特征重要性，并筛选出特征重要性大于 0.008 的分子描述符。以下分别是影响 Caco-2、CYP3A4、hERG、HOB、MN 的特征重要性表：

表 5-1 分子描述符与 Caco-2 的部分特征重要性

变量名称	特征重要性
WPATH	0.2902
ECCEN	0.1578
minaaO	0.0347
maxaaO	0.0344
SaaO	0.0317
MLFER_L	0.0177
MDEC-23	0.0141
MLFER_E	0.0137
TopoPSA	0.0107
WTPT-3	0.0094
minwHBa	0.0092
ETA_BetaP_ns_d	0.0088
minHBa	0.0065
MLFER_S	0.0062
sumI	0.0059
MW	0.0055
minHBd	0.0046
SHother	0.0046
SsCH3	0.0041
MDEC-12	0.0039
BCUTc-1h	0.0038
BCUTc-1l	0.0035
SHBd	0.0034
ETA_Shape_P	0.0034
SaasC	0.0033

表 5-2 分子描述符与 CYP3A4 的部分特征重要性

变量名称	特征重要性
SP-4	0.4048
ETA_Eta_L	0.0591
ETA_dEpsilon_D	0.0343
VP-3	0.0280
SP-3	0.0254
SHBd	0.0234
ETA_Eta	0.0120
ATSc2	0.0113
minHBa	0.0112
ATSc1	0.0109
maxsOH	0.0105
VP-2	0.0075
Zagreb	0.0073
ETA_BetaP_s	0.0065

SP-6	0.0061
minHCsats	0.0060
ETA_Shape_Y	0.0058
SCH-6	0.0056
VP-7	0.0053
minssCH2	0.0052
bpol	0.0052
MDEC-23	0.0047
SP-7	0.0046
VCH-6	0.0045
SCH-7	0.0040

表 5-3 分子描述符与 hERG 的部分特征重要性

变量名称	特征重要性
WTPT-3	0.1612
WTPT-5	0.1366
nHBAcc_Lipinski	0.0828
ETA_Epsilon_1	0.0577
ETA_BetaP_s	0.0500
ETA_EtaP_B_RC	0.0323
maxsCH3	0.0190
FMF	0.0153
TopoPSA	0.0148
mindssC	0.0128
maxsssCH	0.0111
SssCH2	0.0088
ETA_dEpsilon_A	0.0087
MLFER_E	0.0080
ALogp2	0.0071
ETA_Shape_Y	0.0069
minsCH3	0.0068
SssO	0.0066
SsOH	0.0058
nssCH2	0.0051
SCH-7	0.0050
nHBAcc	0.0045
VCH-7	0.0045
SsssCH	0.0045
MLFER_S	0.0045

表 5-4 分子描述符与 H0B 的部分特征重要性

变量名称	特征重要性
BCUTc-1l	0.3006
maxHBint6	0.0174

minHCsatu	0.0126
minHBint6	0.0125
SHsOH	0.0123
SCH-6	0.0114
ETA_BetaP_s	0.0092
maxsCH3	0.0090
maxHsOH	0.0090
FMF	0.0090
nHBint3	0.0088
SdO	0.0087
maxHCsatu	0.0083
hmax	0.0075
minHBint7	0.0074
WTPT-2	0.0072
maxsOH	0.0071
MDEC-33	0.0069
ATSc3	0.0064
ATSc2	0.0060
VP-3	0.0059
MLFER_A	0.0059
ETA_Shape_Y	0.0059
minHsOH	0.0057
SsOH	0.0056

表 5-5 分子描述符与 MN 的部分特征重要性

变量名称	特征重要性
ECCEN	0.2343
VP-0	0.1522
MDEO-11	0.0426
McGowan_Volume	0.0378
bpol	0.0261
maxHsOH	0.0195
WPATH	0.0146
minaasC	0.0109
Kier2	0.0104
hmin	0.0084
SHBint8	0.0084
C1SP3	0.0083
VP-1	0.0078
maxaaCH	0.0067
minssCH2	0.0057
ETA_Shape_Y	0.0057
SsOH	0.0054
minsssN	0.0052

SaasC	0.0048
minHBd	0.0045
FMF	0.0045
LipoaffinityIndex	0.0044
BCUTp-1h	0.0043
maxHBint8	0.0042
WTPT-4	0.0042

筛选出特征重要性大于 0.008 的分子描述符，分别作为主要影响 Caco-2、CYP3A4、hERG、HOB、MN 的自变量。提取了 Caco-2、CYP3A4、hERG、HOB、MN 主要特征变量个数分别为 12 个、11 个、13 个、13 个，12 个，如下图 5-3 分别是对应的可视化柱状图：

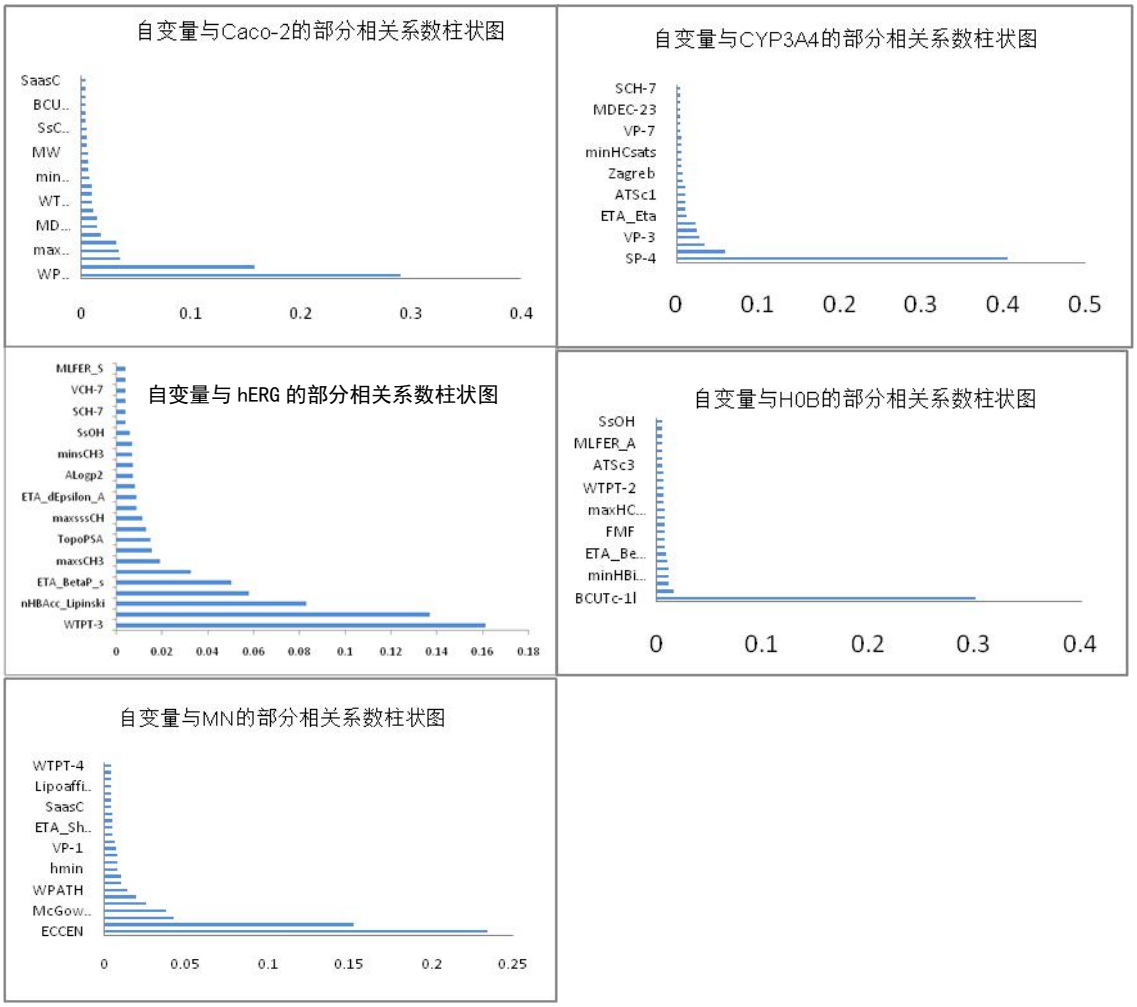

图 5-3 分子描述符与各个因变量之间相关系数

5.4 基于随机森林二分类预测模型

在 5.3 中本文运用随机森林法分别挑选出对 Caco-2、CYP3A4、hERG、HOB、MN 有显著影响的分子描述符，以此确定出了影响 5 种性质的重要变量。接下来，本文将根据筛选好的变量，基于随机森林二分类法对 5 种性质分别建立分类预测模型。

5.4.1 随机森林二分类预测理论介绍

其原理与第二问的随机森林预测等同，区别是第二问采用的是随机森林的数值预测，此问题采用随机森林的二分类预测。

5.4.2 随机森林二分类预测模型建立

随机森林二分类预测模型建立过程与第二问随机森林数值预测模型建立过程一致，只是预测的数据类型不一致，从而不做过多赘述。

5.4.3 随机森林二分类预测结果分析

预测准确率 p 计算公式为：

$$p=\frac{1}{n}\sum_{i=1}^{n}I(\hat{y}_i=y_i) \tag{5.3}$$

$I(\hat{y}_i=y_i)$ 为示性函数，当 $\hat{y}_i=y_i$ 条件成立时，$I(\hat{y}_i=y_i)$ 的值为 1，反之则为 0。

利用随机森林二分类预测模型，预测结果准确度如下表所示：

表 5.6 随机森林二分类预测模型评价表

二者变量	Caco-2	CYP3A4	hERG	HOB	MN
预测准确率	90.91%	90.68%	92.52%	81.11%	91.83%

分析表格发现预测 Caco-2、CYP3A4、hERG、MN 准确率都大于 90%，从而判断出建立随机森林二分类模型对于这些变量的预测比较适用，而对于 HOB 预测准确率小于 90%，对于这个变量随机森林二分类预测模型不适用。

5.5 基于 SVM 支持向量机分类预测模型

在 5.3 中本文运用随机森林法分别挑选出对 Caco-2、CYP3A4、hERG、HOB、MN 有显著影响的分子描述符，以此确定出了影响 5 种性质的重要变量。

5.5.1 基于 SVM 支持向量机分类预测模型的建立

根据筛选好的变量，基于 SVM 支持向量机法对 5 种性质分别建立预测模型。

（1）基于 SVM 支持向量机分类预测模型

SVM 支持向量机分类模型基于训练集合所在的空间，根据样本点所带有的分类标签，寻找一个划分超平面把不用类别的样本区分开, 超平面表达式：

$$w^T x + b = 0 \tag{5.4}$$

在式中，w 为法向量决定超平面的方向；b 为位移决定超平面和原点之间的距离。

模型参数的估计是一个最优化问题，需要求最大值

$$\max_{w,b} \frac{1}{\|w\|} \tag{5.5}$$

其约束条件为

$$s.t. y_i(w^T x_i + b) \geq 1, i = 1,2,...,n \tag{5.6}$$

其中 y_i 为样本标签， x_i 为样本。

通过求解上述最优化问题，得到对应的参数估计，最终预测模型为

$$Sign(w^T x_i + b). \tag{5.7}$$

在式中，若 $w^T x_i + b$ 大于 0 则输出值为 1，反之输出值为 0.

（2）核函数

SVM 模型用于解决线性可分数据集，需要引入核函数概念进而改进模型。当模型自变量在样本空间中线性不可分时，采用核函数将自变量引入更高维度的特征空间中建立模型。此时对应的划分超平面模型变为

$$W^T \theta(x) + b = 0 \tag{5.8}$$

需要求解的最优化问题也随之变为

$$\max_{w,b} \frac{1}{\|w\|} \tag{5.9}$$

其约束条件为

$$s.t.y_i(w^T\theta(x_i)+b)\geq 1, i=1,2,...,n \tag{5.10}$$

对于两个自变量，有

$$\theta(x_i)^T\theta(x_j)=k(x_i,x_j) \tag{5.11}$$

通常称 $k(*,*)$ 称为核函数，大多数模型采用高斯核

$$k(x_i,x_j)=\exp(-\frac{\left\|x_i-x_j\right\|^2}{2\delta^2}) \tag{5.12}$$

最终可以得到预测模型为

$$Sign(w^T\theta(x_i)+b). \tag{5.13}$$

5.5.2 基于支持向量机分类预测模型的求解

利用支持向量机分类预测模型，预测结果准确度如下表所示：

表 5.7 支持向量机二分类预测模型评价表

二者变量	Caco-2	CYP3A4	hERG	HOB	MN
预测准确率	85.81%	89.31%	89.80%	91.85%	85.29%

分析表 5.7 中的数据，可以发现 SVM 支持向量机分类预测模型对于 HOB 性质的预测准确性较高为 91.85%，而对于 Caco-2、CYP3A4、hERG、HOB、MN 性质的预测准确性低于 90%。经过对比分析，认为 Caco-2、CYP3A4、hERG、HOB、MN 变量，基于随机森林的分子描述符特征筛选模型的总体效果要比基于 SVM 支持向量机分类预测模型好。

5.6 分类预测模型的选择及预测

通过对比二种模型预测效果，对于 Caco-2、CYP3A4、hERG、MN 选取随机森林二分类预测模型，对于 HOB 选取支持向量机二分类预测模型，对文件“ADMET.xlsx”的 test 表中的 50 个化合物进行相应的预测，并将结果填入“ADMET.xlsx”的 test 表中对应的 Caco-2、CYP3A4、hERG、HOB、MN 列。预测值如下表 5.8 为：

表 5.8 50 个化合物进行相应的 Caco-2、CYP3A4、hERG、HOB、MN 预测值

序号	Caco-2	CYP3A4	hERG	HOB	MN
1	0	1	1	0	0
2	0	1	1	0	1
3	0	1	1	0	0
4	0	1	1	0	1
5	0	1	0	0	0
6	0	0	1	0	1
7	0	0	1	0	0
8	0	0	1	0	1
9	0	1	1	0	0
10	0	1	1	0	0
11	0	1	1	0	1
12	0	1	1	0	0
13	0	1	1	0	1
14	0	1	1	0	1
15	0	1	1	0	1
16	0	1	1	0	1
17	0	1	1	0	1
18	0	1	1	0	1
19	0	0	1	0	1
20	0	0	0	0	0
21	0	1	1	0	0
22	0	1	1	0	0
23	1	0	0	1	0
24	1	0	0	1	0
25	1	0	1	1	0
26	1	1	1	1	0
27	0	1	1	1	0
28	0	1	1	0	1
29	0	1	1	0	1
30	0	1	1	1	1
31	1	1	1	1	1
32	1	1	1	1	0
33	1	0	1	1	1
34	1	1	1	1	0
35	0	1	1	1	1
36	0	1	0	1	1
37	0	1	0	1	1
38	0	1	1	0	0
39	0	1	1	1	1

40	0	1	1	1	1
41	0	1	1	1	1
42	0	1	1	1	1
43	0	1	1	1	1
44	0	1	1	1	1
45	0	1	1	1	1
46	0	1	1	0	1
47	0	1	1	0	0
48	0	1	1	0	1
49	0	1	1	0	0
50	0	1	1	0	0

6 问题四的分析与求解

6.1 问题分析

针对问题四，题目要求寻找满足五个 ADMET 性质中至少三个性质较好条件，在满足条件下寻找并阐述化合物的哪些分子描述符使得化合物对抑制 ERα 具有更好的生物活性，同时给出这些分子描述符在取值或者处于取值范围。

首先，需要查阅资料文献判断出 IC_{50} 值大于多少，表示该化合物的生物活性几乎没有。其次，需要同时满足生物活性较好、ADMET 性质较好条件，去挑选化合物。这就需要满足双重性质条件下，对数据进行降维，挑选出具有代表性的分子描述符。

其次，查阅到美国 NCI 建议指出，IC_{50} 大于 50uM 可认为该化合物生物活性几乎没有。再通过转化公式可知，pIC_{50} 需要大于 5.3，才认为对应化合物才算有生物活性。然后，设定 pIC_{50} 数值大于 5.3 的条件，在所给予的 1974 个化合物中挑选满足条件的化合物，最后得到 979 个样本化合物数据。

再次，根据 Caco-2、CYP3A4、hERG、HOB、MN 性质，在前面问题假设心脏毒性具有致命性以及遗传毒性具有完全遗传性的条件下，可以将具有遗传毒性或具有心脏毒性的化合物绝对排出在外。于是，满足五个 ADMET 性质中至少三个性质较好条件可以表示为 MN、hERG 取值为 0，其余变量 Caco-2、CYP3A、HOB 共有 7 个可能取值组合。最后根据转换后的条件，当 MN、hERG 取值为 0，可以筛选得到 236 个样本。再根据第二问和第三问，共有六个模型的分子描述符（训练自变量）对模型有显著影响，取六个模型自变量的并集，选出对六个模型有影响的 70 个自变量。

最后，取 Caco-2、CYP3A、HOB 共有 7 个可能取值组合并集，可以得到 97 个样本。对于连续变量，寻找分子描述符的$(Q_1-1.5IQR, Q_3+1.5IQR)$ 作为取值范围；对于非连续变量，寻找出现次数最多的值作为最优取值。其中 $IQR=Q_3-Q_1$，Q_1 为上四分位数、Q_3 为下四分位数。

6.2 分子描述符最优范围优化模型

分子描述符最优范围优化模型的具体算法流程为：

Step1：挑选出满足 pIC_{50} 值大于 5.3 条件的化合物，得到 979 个样本数据。

Step2：将 MN、hERG 取值为 0，筛选化合物，进一步得到 236 个样本数据。

Stsp3：取问题一与问题二中建立的六个模型中的对模型有显著影响分子描述符（训练自变量），取六个模型自变量的并集，选出对六个模型有影响的 70 个分子描述符（自变量）。

Step4：For 循坏遍历 Caco-2、CYP3A4、HOB 的 7 种取值情况

For 记录该情况下对应化合物，并将其存入固定列表

Return 97 个化合物样本

Step5：对于对于连续变量，寻找分子描述符的$(Q_1-1.5IQR, Q_3+1.5IQR)$ 作为值范围；对于非连续变量，寻找出现次数最多的值作为最优取值。其中 $IQR=Q_3-Q_1$，Q_1 为上四分位数、Q_3 为下四分位数。

Step6：得出满足五个 ADMET 性质中至少三个性质较好条件，得到在满足条件下化合物的分子描述符以及同时给出这些分子描述符的取值或者处于取值范围。

6.3 化合物活性的失活临界值

对于药物的筛选，必须要清楚的知道化合物活性的失活的临界值。通过查阅文献资料可知，美国 NCI 认为 IC_{50} 大于 50uM 可认为该化合物生物活性几乎没有，进而通过公式 $pIC_{50}=-\log_{10}\left(IC_{50}\times10^{-9}\right)$，其中 IC_{50} 的单位是 nM，得到 pIC_{50} 需要大于 5.3，才认为对应化合物才算有生物活性。

6.4 依据失活临界值筛选样本

利用 excel 表工具，挑选出满足 pIC_{50} 值大于 5.3 条件的化合物，可以初步得到 979 个样本数据。979 个样本数据，作为下面模型的数据集合。

6.5 依据 ADMET 性质转换变量条件进一步筛选样本

根据 Caco-2、CYP3A4、hERG、HOB、MN 性质，可知当 MN、hERG 均取 1 时，分别代表该化合物具有遗传毒性、该化合物具有心脏毒性。

由于 hERG 取值为 1 时，代表该化合物具有心脏毒性，但是由于不能量化毒性大小，从而假设认为该心脏毒性具有致命的毒性；由于 MN 取值为 1 时，代表该化合物具有遗传毒性，但是由于不能量化遗传性，从而假设认为后代一定会具有该遗传毒性；

由于样本数据的数据量太大，从而再根据第二问和第三问，共有六个模型的分子描述符（训练自变量）对模型有显著影响，取六个模型自变量的并集，选出对六个模型有影响的 70 个分子描述符（自变量）。

题目要求为化合物中五个 ADMET 性质中至少三个性质较好条件，则为保证医疗安全，可以将具有遗传毒性或具有心脏毒性的化合物绝对排出在外。进而，条件转化为：MN、hERG 取值为 0，另外变量 Caco-2、CYP3A4、HOB 可以有 7 中取值方法。如下表 6-1 为所有取值情况：

表 6-1 满足 ADMET 性质条件下的 Caco-2、CYP3A4、HOB 可能取值

序号	Caco-2	CYP3A4	HOB
1	1	0	1
2	0	0	1
3	1	1	1
4	1	0	0
5	0	1	1
6	1	1	0
7	0	0	0

6.6 分子描述符最优取值或最优范围结果

依据算法循坏遍历 Caco-2、CYP3A4、HOB 的 7 种取值情况，可以筛选出满足题目条件的 97 个化合物样本，从而确定了 70 个分子描述符。然后，依据对于连续变量，寻找分子描述符的$(Q_1-1.5IQR, Q_3+1.5IQR)$作为取值范围；对于非连续变量，寻找出现次数最多的值作为最优取值。其中 $IQR=Q_3-Q_1$，Q_1为上四分位数、Q_3为下四分位数。得出满足五个 ADMET 性质中至少三个性质较好条件，得到在满足条件下化合物的分子描述符以及同时给出这些分子描述符的取值或者处于取值范围。分子描述符最优取值或最优范围结果如下 6-2 表：

表 6-2 分子描述符最优取值或最优范围结果

序号	分子描述符	取值范围或定值
0	SaaCH	（12.450，25.898）
1	WPATH	（807,4259.8）
2	ETA_dEpsilon_A	（0.078，0.146）
3	SHBint8	0
4	WTPT-3	(6.370,14.932)
5	WTPT-5	(0,3.407)
6	BCUTp-1h	(11.411,13.109)
7	minHCsatu	(0,0.622)
8	ETA_Eta_L	(4.454,9.660)
9	minsOH	(9.209,10.057)
10	minaasC	(0.163,0.372)
11	MDEC-22	(8.155,30.184)
12	minaaO	0
13	C1SP3	(0.4,6)
14	ATSc2	(-0.273,-0.045)
15	hmin	(-0.404,0.193)
16	SaaO	0
17	ETA_EtaP_B_RC	(0.018,0.030)
18	SdO	(0,10.644)
19	minHBint6	(0,1.114)
20	minwHBa	(-0.335,0.351)
21	ETA_BetaP_ns_d	(0.024,0.057)
22	ETA_Epsilon_1	(0.521,0.591)
23	maxHsOH	(0.444,0.598)
24	nT6Ring	3
25	bpol	(18.432,43.966)
26	SCH-6	(0.192,0.466)
27	Kier2	(6.012,13.645)
28	SHsOH	(0.453,1.139)
29	nHaaCH	(6,13)
30	nHBAcc_Lipinski	(2,5)
31	FMF	(0.292,0.515)
32	nHBint3	1
33	MDEO-11	(0,0.111)
34	SP-4	(6.282,11.130)
35	CrippenLogP	(4.227,6.952)
36	SwHBa	(17.396,33.149)
37	minHBa	(2.105,9.276)
38	MLogP	(3.110,4.320)
39	TopoPSA	(38.770,62.160)
40	mindssC	(0,1.134)

41	McGowan_Volume	(2.150,3.796)
42	MLFER_L	(10.731,18.104)
43	SHBd	(0.467,1.147)
44	n6Ring	4
45	VP-0	(11.788,20.975)
46	MDEC-23	(21.654,38.125)
47	ATSc1	(0.171,0.555)
48	BCUTc-1l	0
49	nC	(18,31)
50	maxaaO	0
51	ATSp4	(2550.625,5052.288)
52	maxHCsatu	(0,0.954)
53	maxHBint6	(0,1.114)
54	ETA_Eta	(14.164,35.992)
55	LipoaffinityIndex	(7.423,13.874)
56	C2SP2	(8,15)
57	ECCEN	(358,1093)
58	VP-3	(3.963,7.412)
59	maxsssCH	(0,0.760)
60	MLFER_E	(1.714,2.940)
61	SssCH2	(0.687,9.934)
62	ETA_BetaP_s	(0.568,0.629)
63	ETA_dEpsilon_D	(0.014,0.046)
64	naaCH	(6,13)
65	SP-5	(5.171,9.458)
66	maxsOH	(9.557,10.134)
67	SP-3	(7.736,13.251)
68	AMR	(87.575,148.868)
69	maxsCH3	(0,2.302)

7 总结

7.1 模型的评价

7.1.1 模型的优点

（1）本文在解决问题时，尝试使用多种办法，采用了多个模型。如针对问题二预测 50 个化合物的生物活性时，本文采用了多元线性回归、双隐层 BP 神经网络、随机森林三个模型进行定量预测 pIC_{50} 值，并且对三种模型的误差大小进行分析比较，从而得到较好的预测结果。

（2）本文在建模之前考虑较为全面。如在建立预测模型之前，根据自变量和因变量散点图的分布来粗略判断它们之间的线性性质，再根据这些性质去建立合适的模型。

（3）本文对不同预测模型的误差大小进行了对比。本文建立了多种类型的预测模型并对其分别计算出 MSE 和 MAPE 的值，通过对比多种预测模型的误差值来选取预测效果最佳的模型。

7.1.2 模型的缺点

（1）本文的预测模型虽然预测效果较好，但由于受到样本条件的约束和实际情况的复杂性影响，对于实际化合物的性质预测依然具有局限性。

（2）由于受到所学专业的限制，本文对于治疗乳腺癌药物的医学机理并未做深入分析，大多数的结论都是基于数据得出的。如果能与医学知识相结合，就能对问题做更全面、深入的分析。

7.2 模型的推广

本文中所建立的针对治疗乳腺癌化合物的生物活性及 ADMET 性质优化模型，可以推广到研究治疗其它疾病中。化合物要想成为药物必须经过严格的筛选和鉴定，本文使用多种方法得到优化模型可以初步判断出化合物的生物活性及 ADMET 性质，将该方法推广到化合物的筛选中可以大大提高寻找合适化合物作为药物的效率。

参考文献

[1] Study of Peak Load Demand Estimation Methodology by Pearson Correlation Analysis with Macro-economic Indices and Power Generation Considering Power Supply Interruption. Journal of Electrical Engineering & Technology, 2017,12(4).

[2] 李晓峰,刘光中.人工神经网络 BP 算法的改进及其应用[J].四川大学学报(工程科学版),2000(02):105-109.).

[3] 熊俊,王士同,潘永惠,包芳.基于惩罚函数泛化的神经网络剪枝算法研究[J].计算机工程,2014,40(11):149-154.

[4] 刘江华,程君实,陈佳品.支持向量机训练算法综述[J].信息与控制,2002(01):45-50.

[5] 张茹玉,叶潘,李月潞,周迎,陈秀奎,刘桦.抗乳腺癌基质金属蛋白酶 9 类抑制剂的分子设计研究[J].成都医学院学报,2021,16(05):571-576.

附录

附录 1 问题一和问题二的代码

1.1 数据预处理

```
import numpy as np
import pandas as pd
import torch
from d2l import torch as d2l
from torch import nn
import pandas as pd
from torch.utils import data
from torch.nn import functional as F
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt

#读取原数据表格
fileName = '/Users/moubaokui/Documents/Molecular_Descriptor.xlsx'
data = pd.read_excel(fileName,sheet_name=0)

#作出所有自变量与因变量的相关系数矩阵,并筛选合适的自变量。
B=data.corr()
corrpIC50=B['pIC50']
index1=corrpIC50[corrpIC50>0.4]
index2=corrpIC50[corrpIC50<-0.4]

def index(x1,x2):
    x=[]
    for i in range(len(x1)):
        x.append(x1[i])
    for i in range(len(x2)):
        x.append(x2[i])
    return x
x=index(index2.keys(),index1.keys())
newdata=data[x]

def createlist(datas):
    Set = set([] )    #创建一个空的不重复列表
    for data in datas:
        x=[]
        x.append(data)
```

```
        Set = Set | set(x) #取并集
    return list(Set)
```

1.2 BP 神经网络预测

```
#排除数据的异常值
def pullerror(data):
    x=[]
    for i in range(len(data.keys())):
        a=data[data.keys()[i]]
        t1=a.quantile(q=0.25)
        t2=a.quantile(q=0.75)
        m=1.5*(t2-t1)
        for j in range(len(a)):
            if a[j]>t2+m or a[j]<t1-m :
                x.append(j)
    return createlist(x)
x2=pullerror(newdata)
realdata=newdata.drop(x2,axis=0)
x1=['C1SP2','minsssN','maxsssN','hmin']

#得到最终训练数据
realdata=realdata.drop(x1,axis=1)

#设置归一化函数，将数据归一化，并粗滤划分训练机和测试机用于检验
def centralization(data):
    centralizationdata=data.apply(lambda x : (x-np.min(x))/(np.max(x)-np.min(x)))
    return centralizationdata
#centraldata=centralization(realdata)
alldata=realdata.iloc[:,0:20];label=realdata.iloc[:,-1]
traindata1=alldata.iloc[100:1100,]
testdata=alldata.iloc[1100:1150,]
trainlabel1=label.iloc[100:1100]
testlabel=label.iloc[1100:1150]
ctraindata1=centralization(traindata1)
ctestdata=centralization(testdata)
ctrainlabel1=(trainlabel1-min(trainlabel1))/(max(trainlabel1)-min(trainlabel1))
ctestlabel=(testlabel-min(testlabel))/(max(testlabel)-min(testlabel))
#转化为 tensor
ctraindata1=torch.tensor(ctraindata1.values,dtype=d2l.float32)
ctestdata=torch.tensor(ctestdata.values,dtype=d2l.float32)
ctrainlabel1=torch.tensor(ctrainlabel1.values,dtype=d2l.float32)
ctestlabel=torch.tensor(ctestlabel.values,dtype=d2l.float32)
```

```
#定义损失函数
def MAE(x1,x2):
    return (abs(x1-x2).sum())/len(x1)
def MSE(x1,x2):
    m=((x1-x2)**2).sum()
    return m/len(x1)
def RMSE(x1,x2):
    return np.sqrt(MSE(x1,x2))

#抽取数据
train_iter = d2l.load_array((ctraindata1,ctrainlabel1.reshape(-1,1)),900)
test_iter = d2l.load_array((ctestdata,ctestlabel.reshape(-1,1)),50)

#设置神经网络
net = nn.Sequential(nn.Linear(20,9),nn.ReLU(),nn.Linear(9,3),nn.ReLU(),nn.Linear(3,1))
for param in net.parameters():
    param.data.normal_()

#神经网络预测加可视化函数
def train_concise(wd):
    loss= nn.MSELoss()
    num_epochs, lr = 900, 0.001
    trainer = torch.optim.Adam([
        {"params":net[0].weight,'weight_decay': wd},
        {'params':net[0].bias}],lr=lr)
    animator = d2l.Animator(xlabel='epochs',ylabel='loss',yscale='log',
                            xlim=[5,num_epochs],legend=['train','test'])
    for epoch in range(num_epochs):
        for X,y in train_iter:
            with torch.enable_grad():
                trainer.zero_grad()#将参数的导数清零，防止自动累加。
                l = loss(net(X),y)
            l.backward()
            trainer.step()
        if (epoch+1)%5==0:
            animator.add(epoch+1,(d2l.evaluate_loss(net,train_iter,loss),
                                  d2l.evaluate_loss(net,test_iter,loss)))
    print("loss：",loss(net(ctestdata),ctestlabel))

loss= nn.MSELoss()
Y=(net(ctestdata)*interval)+minlabel
print(loss(Y,torch.tensor(testlabel.values,dtype=d2l.float32)))
```

```
##数据预测可视化
yp=net(torch.tensor(testdata.values,dtype=d2l.float32))*np.std(testlabel)+np.mean(testlabel)
y=testlabel


# 总体度的分布
x_axis=list(np.arange(0,50))
plt.figure()
plt.plot(x_axis,predictions, label="predict", linestyle="-",c='red')

# advisee 度的折线图分布
plt.plot(x_axis,testlabel,label="real", linestyle="-",c='blue')

# advisor 度的折线图分布
plt.legend()
plt.title("Predict the folding diagram")
plt.xlabel("class")
plt.ylabel("pIC50")
plt.show()
```

1.3 随机森林预测

```
from sklearn.ensemble import RandomForestRegressor
traindata1=alldata.iloc[100:1100,]
testdata=alldata.iloc[1100:1150,]
trainlabel1=label.iloc[100:1100]
testlabel=label.iloc[1100:1150]

#建模
rf=RandomForestRegressor(n_estimators=1000, max_features ='auto',random_state=50)
rf.fit(traindata1,trainlabel1)
predictions=rf.predict(testdata)
print(MSE(predictions,testlabel))
```

附录 2 问题三的代码

2.1 多元线性回归

```
import statsmodels.api as sm
from sklearn.linear_model import LinearRegression
def standar(data):
    standdata=data.apply(lambda x : (x-np.mean(x))/(np.std(x)))
    return standdata
standardata=standar(realdata)
model=sm.OLS(standardata['pIC50'], sm.add_constant(standardata.iloc[:,0:19])).fit()
model=LinearRegression()
```

```
model.fit(standardata.iloc[:,0:19],standardata['pIC50'])

print(model.predict(standardata.iloc[:,0:19]))
import numpy as np
import pandas as pd
from statsmodels.stats.outliers_influence import variance_inflation_factor
import torch
from d2l import torch as d2l
from torch import nn
import pandas as pd
from torch.utils import data
from torch.nn import functional as F
from sklearn.model_selection import train_test_split
from sklearn.datasets import load_boston
from sklearn.ensemble import RandomForestRegressor
```

2.2 随机森林挑选自变量

```
#读取数据
fileName = '/Users/moubaokui/Documents/Molecular_Descriptor1.xlsx'
data = pd.read_excel(fileName,sheet_name=0)
alldata=data.iloc[:,1:730];label1=data['Caco-2']
label2=data['CYP3A4'];label3=data['hERG']
label4=data['HOB'];label5=data['MN']

#挑选 Caco-2 的有效自变量
X=alldata;Y=label1
name=X.keys().tolist()
rf = RandomForestRegressor()#n_estimators=500max_features ='auto',random_state=50
print(rf.fit(X, Y))
liename=sorted(zip(map(lambda x: round(x, 4), rf.feature_importances_), name), reverse=True)
listname=liename[0:25]
print(listname)

#通过观察，挑出最具影响的自变量
realname=listname[0:12]
dataname=pd.DataFrame(realname)[1]
dataname=dataname.tolist()
realdata=data[dataname]

#筛选异常值函数
def createlist(datas):
    Set = set([] )     #创建一个空的不重复列表
    for data in datas:
        x=[]
```

```
            x.append(data)
            Set = Set | set(x) #取并集
        return list(Set)
def pullerror(data):
        x=[]
        for i in range(len(data.keys())):
            a=data[data.keys()[i]]
            t1=a.quantile(q=0.25)
            t2=a.quantile(q=0.75)
            m=1.5*(t2-t1)
            for j in range(len(a)):
                if a[j]>t2+m or a[j]<t1-m :
                    x.append(j)
        return createlist(x)
x2=pullerror(realdata)
realdata1=realdata.drop(x2,axis=0)
Y1=Y.drop(x2,axis=0)

#筛选训练数据和测试数据
train_data,test_data = train_test_split(realdata1,random_state=1, train_size=0.7, test_size=0.3)
train_label,test_label = train_test_split(Y1,random_state=1, train_size=0.7, test_size=0.3)

#预测函数
def tranform(y,ally):
        x=[]
        k=(ally.sum())/len(ally)
        for i in range(len(y)):
            if y[i]>=k:
                x.append(1)
            else:x.append(0)
        return x
def accuracy(ry,y):
        a=0;n=len(ry)
        for i in range(n):
            if ry[i]==y[i]:
                a+=1
        return a/n

#随机森林训练数据
rf1 = RandomForestRegressor()
rf1.fit(train_data, train_label)
y=rf1.predict(test_data)
ry=tranform(y,Y)
print(accuracy(ry,list(test_label)))
```

2.3 SVM 支持向量机预测

```
from sklearn.svm import SVC
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix
from sklearn.metrics import accuracy_score
from sklearn import svm

classifier = svm.SVC()
classifier.fit(train_data,train_label.ravel())
pre_train = classifier.predict(train_data)
pre_test = classifier.predict(test_data)
print('train:',accuracy_score(pre_train,train_label))
print('test:',accuracy_score(pre_test,test_label))

#预测最终 test
tdata=pd.read_excel(fileName,sheet_name=1)[dataname]
finall_y=tranform(rf1.predict(tdata),Y)
excel_y=pd.DataFrame(finall_y)
print(excel_y)
```

附录 3 问题四的代码

```
import pandas as pd
import numpy as np
index1=['AMR', 'nC', 'ATSp4', 'BCUTp-1h', 'C2SP2', 'SP-5', 'CrippenLogP','nHaaCH', 'naaCH',
'SwHBa', 'SaaCH', 'minsOH', 'maxHsOH', 'maxsOH', 'LipoaffinityIndex', 'MLogP', 'MDEC-22',
'MDEC-23', 'n6Ring', 'nT6Ring',]
hERG=['ECCEN','VP-0','MDEO-11','McGowan_Volume','bpol','maxHsOH','WPATH','minaasC','K
ier2','hmin','SHBint8','C1SP3']
Caco_2=['WPATH','ECCEN','minaaO','maxaaO','SaaO','MLFER_L','MDEC-23"MLFER_E','Topo
PSA','WTPT-3','minwHBa','ETA_BetaP_ns_d']
CYP3A4=['SP-4','ETA_Eta_L','ETA_dEpsilon_D','VP-3','SP-3','SHBd','ETA_Eta','ATSc2','minHB
a','ATSc1','maxsOH']
MN=['WTPT-3','WTPT-5','nHBAcc_Lipinski','ETA_Epsilon_1','ETA_BetaP_s','ETA_EtaP_B_RC',
'maxsCH3','FMF','TopoPSA','mindssC','maxsssCH','SssCH2','ETA_dEpsilon_A','MLFER_E']
HOB=['BCUTc-1l','maxHBint6','minHCsatu','minHBint6','SHsOH','SCH-6',
'ETA_BetaP_s','maxsCH3','maxHsOH','FMF','nHBint3','SdO','maxHCsatu']
def createlist(x1,x2):
    Set = set(x1)     #创建一个空的不重复列表
    Set = Set | set(x2) #取并集
    return list(Set)
name1=createlist(index1,hERG);name2=createlist(name1,Caco_2)
```

```
name3=createlist(name2,CYP3A4);name4=createlist(name3,MN)
name5=createlist(name4,HOB)

data=pd.read_excel('/Users/moubaokui/Documents/excel4.xlsx')
newdata=data[name5]
y1=data.iloc[:,-3];y2=data.iloc[:,-4];y3=data.iloc[:,15]
def elect(a,b,c):
    x=[]
    for i in range(len(y1)):
    if y1[i]==a and y2[i]==b and y3[i]==c:
    x.append(i)
 return x

x1=elect(1,0,0);x2=elect(0,0,0)
x3=elect(1,1,0);x4=elect(1,0,1)
list1=createlist(x1,x2);list2=createlist(list1,x3);list3=createlist(list2,x4)

NBdata=newdata.iloc[list3,:]
number=np.zeros((70,2))
for i in range(len(NBdata.keys())):
      c=NBdata.iloc[:,i]
      a=c.quantile(q=0.15)
      b=c.quantile(q=0.85)
      number[i,0]=a
      number[i,1]=b
pd.DataFrame(number).to_excel('/Users/moubaokui/Documents/excel0.xlsx')
```