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		浙江工商大学
参赛队号		**21103530003**
队员姓名	**1.**	刘晓
	2.	邬治彬
	3.	毛安琪

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多目标优化建模

摘　　要：

乳腺癌是目前致死率比较高的癌症，为了提高乳腺癌药物的效果，减少患者的痛苦，提高人们的生活质量，本文通过建立模型对药物效果的影响因素进行分析，并得出相关的预测结果。

对于问题一，通过数据清洗方法，利用 python 编程，查看数据无缺失值，并用箱型图选择性画出异常数据并将标准差为零的特征处理掉，通过孤立森林算法将数据中的 99 条异常数据剔除。利用 MIC，Catboost，RF 三种方法进行特征筛选，并将三种方法筛选的特征按照顺序加权，得到对生物活性更具影响的变量。

对于问题二: 首先通过 XGBoost 进行特征选择，取前 20 个重要变量分别利用 XGBoost,RF 进行预测，同时计算两模型的各项误差 MSE ,MAE,RMSE, 而后通过遗传算法对两模型进行最优参数的调节，得到 GA-XGBoost 和 GA-RF 模型及其各项误差，通过比较得到调参后的模型优于原模型，最后再用遗传算法确定 GA-XGBoost 和 GA-RF 两模型权重，得到组合预测模型，由结果可以明显看出组合预测模型要优于 GA-XGBoost 和 GA-RF。

对于问题三，建立了两种分类预测模型，首先用逻辑回归对 ADMET 的五种化合物进行预测，而后分别用 XGBoost RF 进行变量选择，将得到的特征用于 XGBoost，RF, GA-XGBoost, GA-RF 以及 GA-XGBoost-RF 组合模型中，选择效果更好的模型预测出 ADMET 的五种性质。

对于问题四，利用问题二建立的多项式模型以及第三问建立的逻辑回归模型建立优化模型，并根据分子的取值范围，求出最优解。

关键字： ERα 生物活性　变量选择　GA-XGBoost-RF　组合预测模型　多目标规划

目录

1 问题重述及技术路线图

1.1 问题重述

乳腺癌的治疗研究一直备受各界关注，而 ERα 则被认为是治疗乳腺癌的重要靶标。能够成为候选药物的化合物，不仅得有良好的生物活性，还得在人体内有良好的 ADMET 性质。本文仅考虑化合物的 Caco-2、CYP3A4、hERG、HOB 和 MN 这 5 种 ADMET 性质。围绕 ERα，本文依次提出如下问题：

问题一：根据所提供的 1974 个化合物的 729 个分子描述符（变量）进行变量选择，按照变量对 ERα 生物活性（pIC_{50}）影响的重要程度从高到低排序，并列出前 20 个。

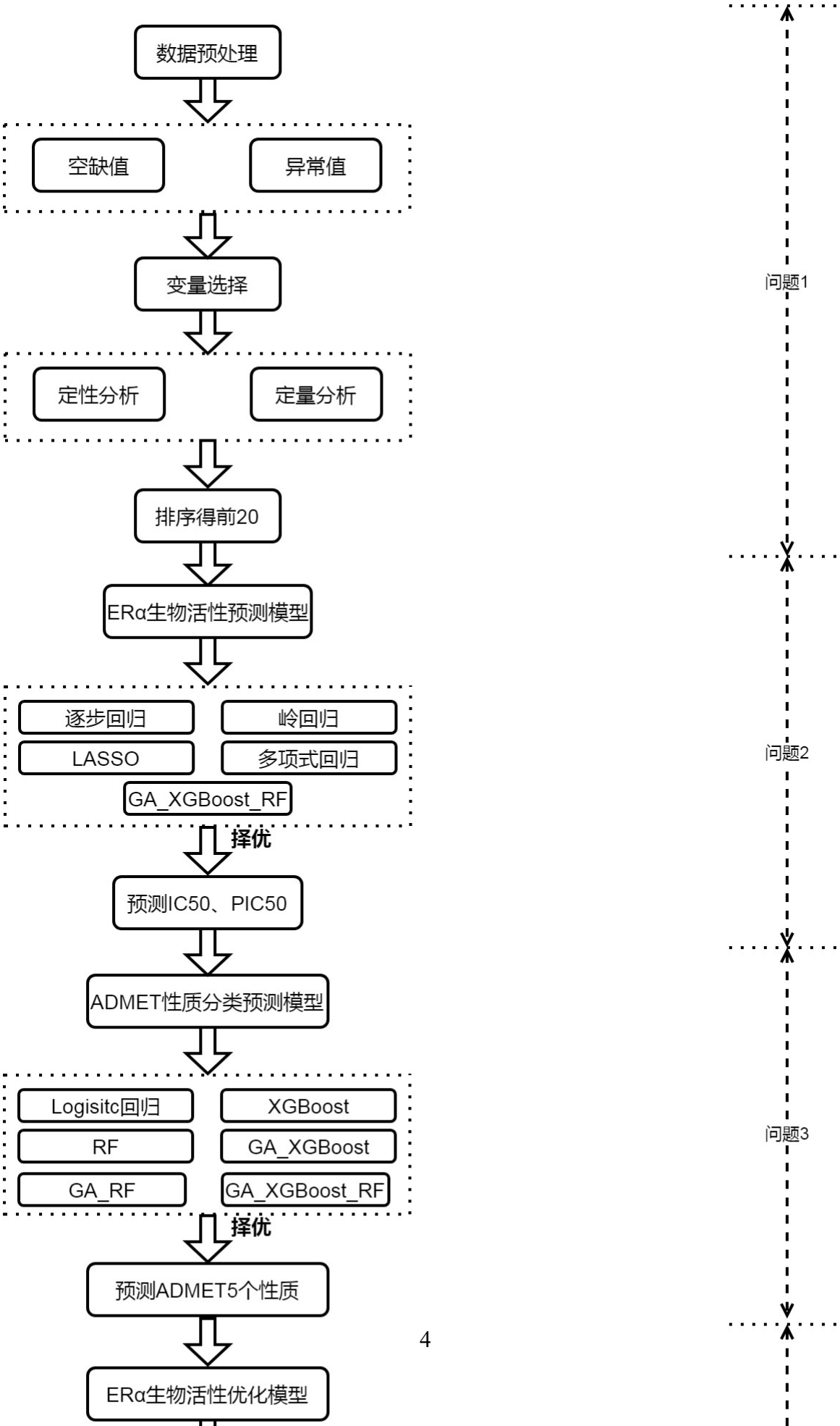
问题二： 筛选不多于 20 个变量，建立 pIC_{50} 的预测模型，并对给定的 50 个化合物的 IC_{50} 值和 pIC_{50} 值进行预测。

问题三： 利用提供的 ADMET 数据，分别建立 Caco-2、CYP3A4、hERG、HOB、MN 的分类预测模型，并对给定的 50 个化合物的这 5 个性质进行预测。

问题四： 求出哪些变量取何值或范围时能够使 pIC_{50} 值更大，同时有更好的 ADMET 性质（最少保证有三个性质较好）。

1.2 技术路线图

全文技术路线图如下：


数据预处理
空缺值
异常值
变量选择
定性分析
定量分析
排序得前20
ERα生物活性预测模型
逐步回归
岭回归
LASSO
多项式回归
GA_XGBoost_RF
择优
预测IC50、PIC50
ADMET性质分类预测模型
Logisitc回归
XGBoost
RF
GA_XGBoost
GA_RF
GA_XGBoost_RF
择优
预测ADMET5个性质
ERα生物活性优化模型
问题1
问题2
问题3

2 模型假设

1. 假设异常值处理后的数据是有效可信的；
2. 假设无数据集以外对 ERα 活性影响很大得变量；
3. 假设模型所算出来的误差对预测结果没有明显影响。

3 符号说明

3.1 符号说明

符号	意义
y_0	pIC50
y_1	Caco-2
y_2	CYP3A4
y_3	hERG
y_4	HOB
y_5	MN

4 问题一：ERα 生物活性变量选择组合模型的建立与求解

4.1 问题一分析

首先，结合问题背景对数据进行预处理，需特别考虑到各分子描述符的实际含义。然后，结合数据特征，选择能够将变量的重要性进行排序的变量选择的方法建立模型。其中最大信息系数法（MIC）、随机森林 (RF) 和 CatBoost 既能处理本文的数据又可以得出变量的重要性排序，但鉴于各方法的优缺点分析，可以将三种方法进行组合，从而建立基于 MIC、RF 和 CatBoost 的变量选择组和模型。最后针对 729 个分子描述符（变量），得出各变量的综合得分，按照综合得分从高到低排序，由于问题背景的特殊性，我们不能仅靠定量分析，还需结合问题背景与实际情况综合考量，所以可以将定量分析与定性分析相结合，最终列出前 20 个最重要的变量。

技术路线图如下：

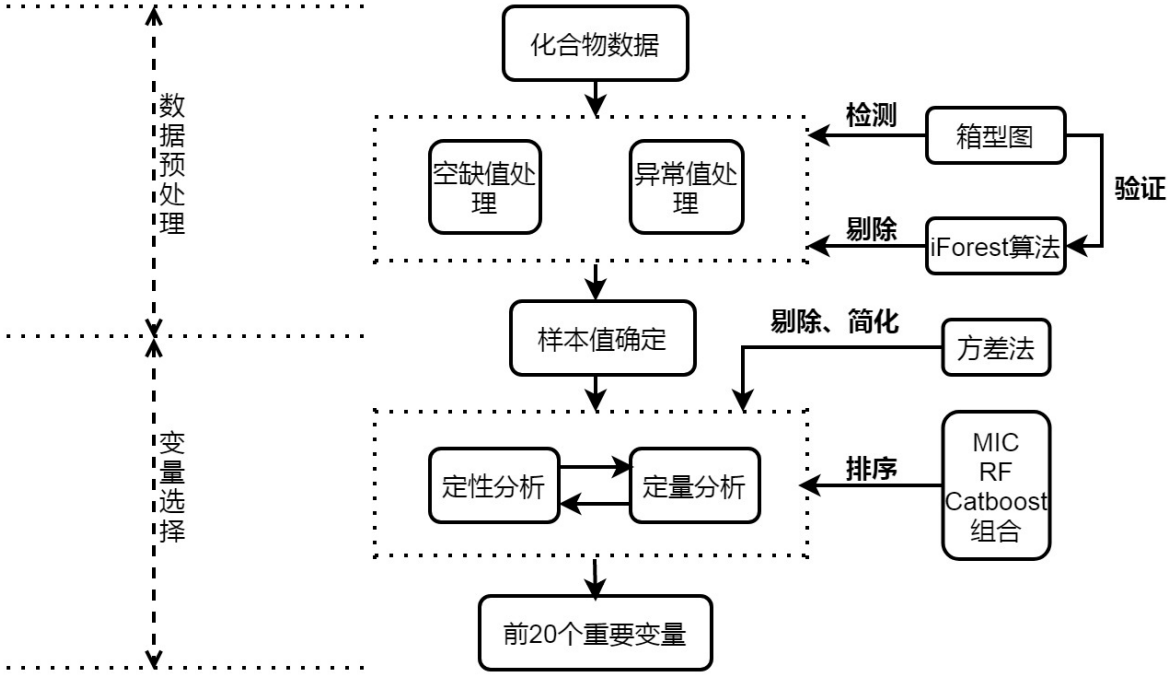


图 4.2　问题一技术路线图

4.2 数据预处理

4.2.1 缺失值处理

通常人为或机器等原因会造成数据的缺失，从而使系统丢失有用的信息，因此在对数据进行分析操作之前，需要先检查数据集是否有缺失值情况并做相应的处理。

利用 python 软件检查每个文件的数据，结果发现各文件中均没有缺失值，因此不需要缺失值处理。

4.2.2 异常值处理

假如数据中有异常值，特别是存在偏离较大的离群点的时候，将会给分析以及建模带来一定的误差，因此在对数据进行分析操作之前，需要先检查数据集是否有异常值情况并做相应的处理。

比较常用的异常值检测方法有 3σ 准则或 $Z-score$ 法，可是这类方法以正态分布为前提条件。而本文的数据有一些不服从正态分布，所以本文选择对数据分布没有要求的箱型图以及孤立森林（iForest ）算法相结合的方法，对数据进行异常值检测。鉴于很多分子描述符就算异常也有其实际意义，不便对其异常值进行替换，所以本文对异常数据直接删除处理。

(1) 箱型图

箱型图是一个对数据分布没有要求的异常值检测方法。如图4.3：

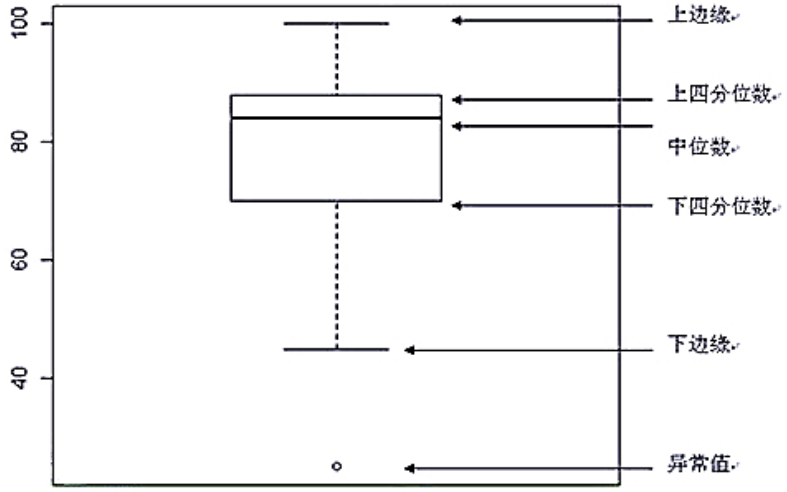


图 4.3　箱型图图示

记 Q_3 为上四分位数,Q_1 为下四分位数，IQR 为统计内距，即 $IQR = Q_3 - Q_1$，则上边缘可表示为 $Q_3 + 1.5IQR$, 下边缘表示为 $Q_1 - 1.5IQR$，异常点指的就是处于上边缘或下边缘之外的点。

(2)iForest 算法

iForest 算法是基于 Ensemble 的异常检测法，因此具有线性的时间复杂度，且精准度较高；在训练过程中，不需要记得距离、密度等指标，因此运算速度较快。iForest 由 t 个孤立树组成，每个孤立树是一个二叉树结构，实现步骤为：

1、随机选择 ψ 个样本点作为子样本，放入树的根节点。

2、随机指定一个维度，在当前节点数据中指定维度的最大值和最小值之间随机产生一个切割点 p。

3、由 p 切割点产生的超平面将节点数据空间划分为 2 部分，将小于 p 的数据放在左分支，否则放在右分支。

4、循环 2、3 步，直到分支节点中只保留一个数据或者分支节点达到限定高度。

接下来计算异常分数 s，用来评估测试数据。异常得分越接近于 1，其越异常。

$$s\left(x,\psi\right) = 2^{-\frac{E(h(x))}{c(\psi)}}$$

其中 $h\left(x\right)$ 为 x 在每棵树的高度，$c\left(\psi\right)$ 为给定样本数 ψ 时路径长度的平均值，用来对样本 x 的路径长度 $h\left(x\right)$ 进行标准化处理。

首先处理文件“Molecular_Descriptor.xlsx”中的数据。经箱型图检测，该文件中有多处异常值，iForest 算法处理后，剔除掉 99 个化合物的数据，将处理后的数据再做一次箱型图，部分结果如下图：

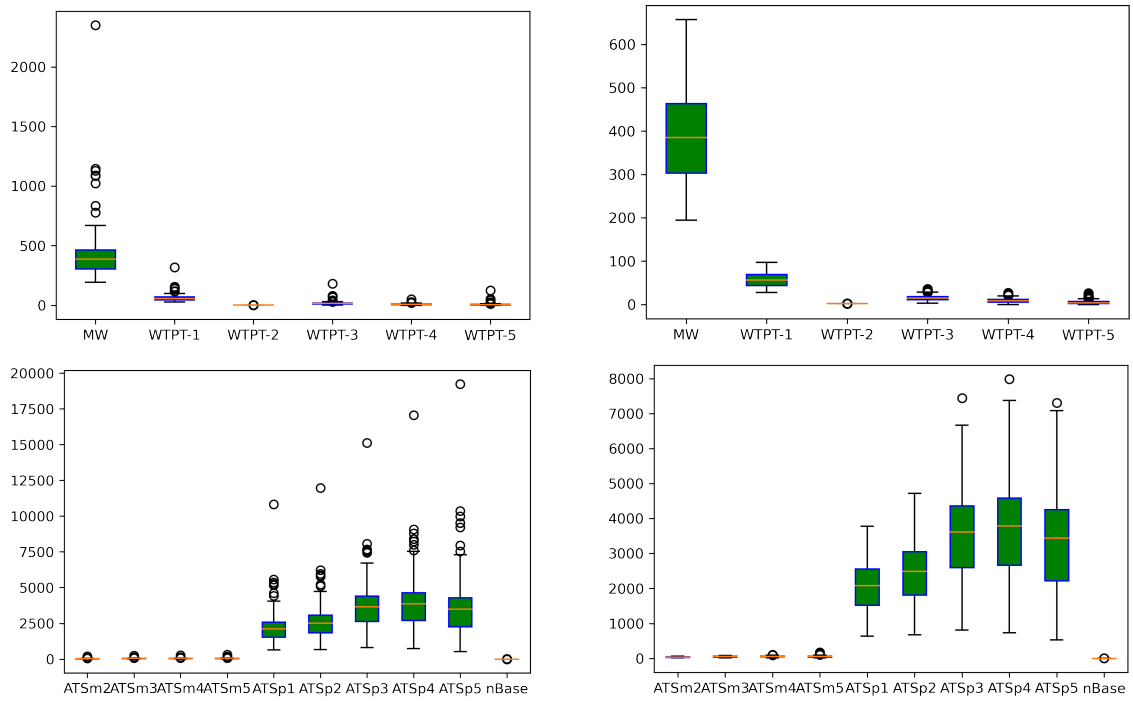


图 4.4　异常值处理前后对比图

由图可知，一部分变量的箱线图在异常值处理之后变化比较明显，如：MW、WTPT-1、ATSP1、ATSP2 等，这表明异常值处理后，这些变量的分布更加集中，将更加有利于数据分析以及建模。

处理文件“ERα_activity.xlsx”中的数据时发现第二列 IC_{50} 中有异常值，但第三列 pIC_{50} 中并没有，应该是因为 pIC_{50} 是 IC_{50} 的负对数，所以缓和了异常值。鉴于之后用 pIC_{50} 表示生物活性值并参与预测，IC_{50} 中的异常值便不需处理了。而其他文件未发现异常情况。

4.3 数据初始化

有些方法在用之前需要初始化数据以方便求解。本文会用到的数据初始化方法有归一化和标准化。

(1) 数据归一化处理

归一化：把数据转换成 (0□1) 或者 (−1□1□ 的数。归一化的提出主要是为了方便处理数据，将数据映射到一定范围内处理，能够更加便捷快速。并且将有量纲的表达式变成无量纲的，能够使不同量级或单位的指标便于比较和加权。

比较常用的是 0-1 归一化，其表达式为：

$$x' = \frac{(x - x_{\min})}{(x_{\max} - x_{\min})}$$

(2) 数据标准化处理

标准化：标准化后每个特征的均值都变为 0、标准差变为 1，该方法被广泛使用在许多算法中 (如：Logistic 回归、支持向量机、K 近邻和类神经网络)。

Z-score 规范化表达式为：

$$x' = \frac{(x - \mu)}{\sigma}$$

我们将数据分别进行 0-1 归一化和标准化备用，以便之后不同算法的使用。当然，还有一些算法不需要数据初始化，比如 RF 算法，RF 是基于决策树的算法，所以对数据初始化就没有要求。

4.4 基于 MIC、RF 和 CatBoost 的变量选择组和模型

数据预处理后，我们一般从两方面考虑是否选择该变量：

其一，变量的发散程度：若样本的某个变量的方差越接近于 0，则该变量发散程度越低，那么我们认为该变量无法区分该样本。

其二，变量与目标的相关性：与目标相关性越高的变量越应该被选择。

数据预处理完成之后，我们一般从两方面考虑是否选择该变量：其一，变量的发散程度：若样本的某个变量的方差越接近于 0，则该变量发散程度越低，那么我们认为该变量无法区分该样本。其二，变量与目标的相关性：与目标相关性越高的变量越应该被选择。

常见的变量选择方法有过滤式、包裹式及嵌入式选择法。其中过滤式变量选择法直接用统计方式对变量打分，分数越高，价值越大，且其未涉及学习模型，故选择速度较快。但是该方法仅独立考虑每个变量与目标变量间的相关性，忽略了不同变量间的关联性及组合效果。

包裹式变量选择法，是通过机器学习模型、评测性能指标来选择或排除变量，通常情况下，其效果要优于过滤式变量选择法。它可以直接对特定学习器进行优化，且考虑变量间的关联性，但是其选出的变量通用性往往较弱。当遇到大规模数据时，运算时间较长。

嵌入式变量选择法，它是将变量选择过程和模型训练过程结合起来，从而快速找到最佳变量集合。其可以考虑到不同变量组合产生的组合效果来更好的评价特征重要性，但是它的运算时间也比较长。

通过各方面筛选，最大信息系数法（MIC）、随机森林 (RF) 和 CatBoost 既能处理本文的数据又可以得出变量的重要性排序，但鉴于各方法的优缺点分析，可以将三种方法进行组合，从而建立基于 MIC、RF 和 CatBoost 的变量选择组和模型。

4.4.1 方差法

当变量的方差为 0 时，我们认为这一变量无法用于区分样本，这些变量一定不会被选择。所以为了减少计算，我们先用方差法做变量筛选，将方差为 0 得变量直接删除。

4.4.2 最大信息系数法

最大信息系数是以信息论中的互信息为基础的，所以我们首先引入信息论中信息熵、条件熵和互信息量的概念及理论。

信息论是衡量变量不确定性的一种方法，在机器学习的算法中，通常将不确定性小的特征作为最优特征，但是如何将变量中的信息量化之一问题仍未解决。基于此，shannon 于 1948 年提出信息熵的概念，将变量的不确定性通过数值反映出来。

(1) 信息熵

信息熵将变量的不确定性用数值的形式反映出来，信息熵越大则表明特征的不确定性越大。基于离散信源的信息熵的公式为：

$$H(X) = -\sum p(x_i)\log(p(x_i)), i = 1, 2, ..., n$$

其中 X 表示随机变量，信源概率为 $p(x_i)$,$i = 1, 2, ..., r$。

基于上式计算出的信息量为：

$$I(x_i) = -\log(p(x_i))$$

其中信息量 Ix_i 越高，$p(x_i)$ 越小，不确定度越高。

(2) 条件熵

设 X_1, X_2 是两个离散型随机变量，随机变量 X_1 给定的条件下随机变量 X_2 的条件熵 $H(X_2|X_1)$ 表示在已知随机变量 X_1 的条件下随机变量 X_2 的不确定性。公式推导如下：

$$H(X_2|X_1) = \sum_{x_1 \in X_1} p(X_1)H(X_2|X_1 = x_1) = -\sum_{x_1 \in X_1}\sum_{x_2 \in X_2} p(X_2|X_1)\log(p(X_2|X_1))$$

(3) 互信息量

变量 X_1，X_2 的信息熵分别是 $H(X_1), H(X_2)$,X_1，X_2 联合分布的信息熵为 $H(X_1, X_2)$。如果 X_1，X_2 相互独立，则

$$H(X_1, X_2) = H(X_1) + H(X_2)$$

互信息计算公式如下：

$$I(X_1, X_2) = H(X_1) + H(X_2) - H(X_1, X_2)$$

(4) 特征选择的最大信息系数法步骤:

设已知有包含 k 个类别的样本数据集，且这 k 个类别中的每类有 m 个样本，每个样本点包含 N 个特征 $\{x_1{}^{(1)},...,x_m{}^{(1)},...,x_1{}^{(k)},...,x_m{}^{(k)}\}$，其中 $x_i{}^{(k)}=(x_{i,1},...,x_{i,N})$ 表示类别 k 的第 i 个样本的特征组成的向量。

依据单独最优特征组合搜索法的观点将有效性判断值最大的前 n 个变量，从全部变量中筛选出来。将 MIC 作为有效判据，用于单独最优特征组合搜索法中做变量选择，步骤为：

1、从每个类型变量中分别随机选取 $y^{(1)},y^{(2)},...,y^{(k)}$ 作为对照样本，其余为训练样本；

2、将对照样本的第 j 个变量与 $m-1$ 个训练样本的第 j 个变量一一对应，k 个类别一共构成与变量 j 有关的 $k(m-1)$ 个变量对；

3、求出步骤 2 中得到的变量对的 MIC，N 个变量共得到个 N 个 MIC；

4、将全部变量的 MIC 值排序，选择 MIC 值最大的前 n 个变量。

4.4.3 随机森林

随机森林（RF）是由 Leo Breiman 于 2001 年提出的将决策树中 CART 算法和 Bagging 算法相结合的一种新算法，它利用 bootsrap 重采样方法从原始样本中抽取多个样本，对每个 bootsrap 样本进行决策树建模，再通过多棵决策树的组合，最终以投票的方式得出选择结果 [**?**]。

随机森林中某个变量重要性计算步骤如下：

1、使用每颗决策树相应的 OOB（袋外数据）数据来计算它的袋外数据误差，记为 err_{OOB1}；

2、随机地将噪声干扰加入袋外数据 OOB 所有样本的变量 X 中，再次计算其袋外数据误差，记为 err_{OOB2}；

3、变量 X 的重要性 $=\frac{\sum(err_{OOB2}-err_{OOB1})}{N}$（其中 N 表示，随机森林中有 N 棵树）

若某个变量随机加入噪声之后, 袋外数据的准确率大幅度降低, 则说明这个变量对样本的分类结果影响很大, 也就是说它比较重要，我们要将其保留。

4.4.4 CatBoost

CatBoost 是以对称树为基学习器的一种参数较少、支持类别型变量和高准确性的 GBDT 框架。它由 Categorical 和 Boosting 组成，其在处理类别型变量时，首先计算其出现的频率，再加上超参数生成新的数值型变量。此外，它使用组合类别变量，充分利用变量间的联系，丰富变量维度。

对于低维类别型变量，一般采用 One—hot 方法将变量转化为数值型。对于高维类别型变量，通常使用目标变量统计（Target Statistics）进行分组，目标变量统计用于估计每个类别的目标变量的期望值。其可以通过对目标变量统计的数值型特征的阈值设置，基于对数损失、基尼系数或均方差，得到一个对于训练集而言可以将类别一份为二的所有可能

划分中最优的那个。

4.4.5 方法组合

按照以上三个得到各变量的重要程度并进行排序，然后针对每一个变量取三个排名的均值，按从小到大再次排序，最小的就是综合排名第一，依此类推，得出各变量的综合排序。

4.5 模型求解

使用 Python 对数据处理发现，有 214 个变量方差为 0，故直接将这些变量筛除。接下来分别使用 MIC、RF 和 CatBoost 这三种方法对 515 个变量进行重要性打分并排序。

用 MIC 求解的部分排序效果如表4.1 所示：

表 4.1　MIC 得分的部分排序

名称	得分	排序
BCUTc-1l	0.3749	1
MDEC-23	0.3942	2
BCUTc-1h	0.3210	3
MLogP	0.3418	4
WTPT-5	0.3325	5
SsOH	0.3293	6
minHsOH	0.3292	7
minsOH	0.3189	8
SHsOH	0.3427	9
LipoaffinityIndex	0.3579	10

用 RF 求解的部分排序效果如表4.2 所示：

表 4.2　随机森林得分的部分排序

名称	得分	排序
MDEC-23	26.6688	1
C1SP2	2.7006	2
minHsOH	2.5993	3
BCUTc-1l	3.2120	4
maxHsOH	2.5262	5
minsOH	1.7138	6
nC	1.1050	7
maxssO	2.5133	8
LipoaffinityIndex	0.4977	9
nHBAcc	0.8443	10

此次排序 RF 的稳定性评价得分 0.7837。用 CatBoost 求解的部分排序效果如表4.3 所示：

表 4.3　Catboost 得分的部分排序

名称	得分	排序
MDEC-23	3.6293	1
maxHsOH	1.0913	2
ATSp5	3.3848	3
C1SP2	1.5806	4
LipoaffinityIndex	1.7237	5
MLFER_A	2.3081	6
BCUTp-1h	2.6902	7
maxsOH	1.6344	8
nsOH	0.6107	9
minsOH	0.5165	10

求出变量在 3 个模型中排名的平均数，再将排名平均数进行排序，得到最终的重要性排名。取前 35 个如表4.4所示：

表 4.4　变量综合排序

	MIC 排序	RF 排序	Catboost 排序	综合排序
MDEC-23	2	1	1	1
BCUTc-1l	1	4	15	2
maxHsOH	14	5	2	3
minHsOH	7	3	12	4
minsOH	8	6	10	5
LipoaffinityIndex	10	9	5	6
ATSp5	11	14	3	7
nC	15	7	11	8
SsOH	6	25	29	9
BCUTc-1h	3	18	42	10
MLogP	4	35	26	11
C1SP2	62	2	4	12
SHsOH	9	43	23	13
MLFER_A	49	24	6	14
WTPT-5	5	34	43	15
maxssO	68	8	14	16
BCUTp-1h	64	54	7	17
MDEO-12	86	11	32	18
nHBAcc	102	10	20	19
maxsOH	13	115	8	20
SPC-6	107	17	21	21
hmin	41	42	71	22
TopoPSA	58	22	74	23
gmax	40	82	35	24
VC-5	93	21	48	25
SHBint6	127	19	19	26
SHBint10	103	27	46	27
ATSc2	128	12	37	28
ATSc4	110	33	34	29
MDEC-33	113	23	41	30
Kier3	54	29	95	31
CrippenLogP	79	51	69	32
MDEC-22	20	61	119	33
mindssC	123	67	17	34
XLogP	162	30	24	35

通过查找文献资料发现有几个重要的描述符，如 MlogP、XlogP、TopoPSA、nC 和 MW 等，而除 MW 外其余几个重要指标在表4.4中排序都很靠前，从而也从实际意义上验证了基于 MIC、RF 和 CatBoost 的变量选择组和模型的有效性。各方资料表示，nC 和 MW 应该有高度线性关系，所以我们画出了 nC 和 MW 的散点图4.5：

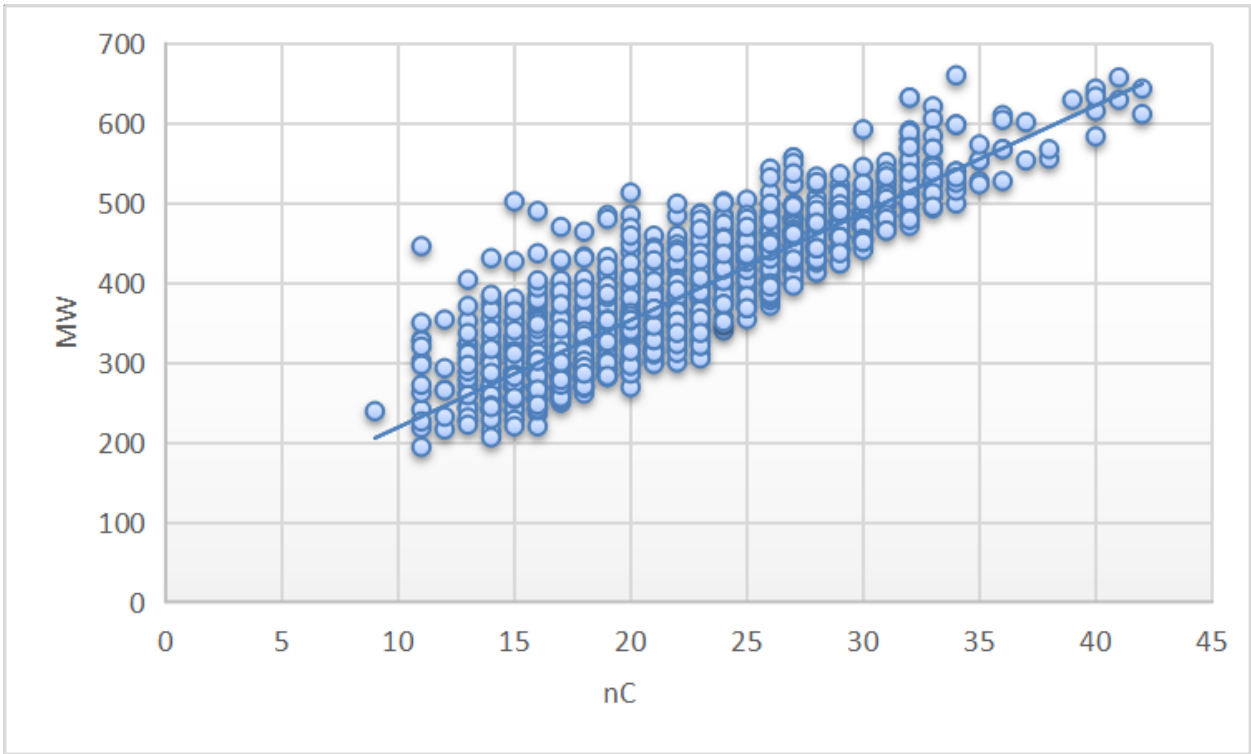


图 4.5　nC 和 MW 散点图

从图中可以很明显的发现，nC 和 MW 高度线性相关，所以变量选择时只需选其中一个就可以，而 nC 本就在综合评分排名中位于第 8，因此最终不考虑将 MW 提至最重要的 20 个指标中。

资料显示 C1SP2 和 nHBAcc 这两个变量并不明显重要，综合考虑将其剔除前 20，而其余变量的排名依然按照原顺序依次排序，最后将原处于 23 名的 TopoPSA 和 35 名的 XLogP 分别放在第 19 和 20。

经过定性分析与定量分析相结合的方式，我们最终得出前 20 的重要变量，并将其依次记为 $x_1, x_2, ..., x_{20}$，如表4.5：

表 4.5　变量重要性综合排序

综合排序	符号	变量	含义
1	x_1	MDEC-23	Molecular distance edge between all secondary
2	x_2	BCUTc-1l	nhigh lowest partial charge weighted BCUTS
3	x_3	maxHsOH	Maximum atom-type H E-State: -OH
4	x_4	minHsOH	Minimum atom-type H E-State: -OH
5	x_5	minsOH	Minimum atom-type E-State: -OH
6	x_6	LipoaffinityIndex	Lipoaffinity index
7	x_7	ATSp5	ATS autocorrelation descriptor, weighted by polarizability
8	x_8	nC	Number of carbon atoms
9	x_9	SsOH	Sum of atom-type E-State: -OH
10	x_{10}	BCUTc-1h	nlow highest partial charge weighted BCUTS
11	x_{11}	MLogP	Mannhold LogP
12	x_{12}	SHsOH	Sum of atom-type H E-State: -OH
13	x_{13}	MLFER_A	Overall or summation solute hydrogen bond acidity
14	x_{14}	WTPT-5	Sum of path lengths starting from nitrogens
15	x_{15}	maxssO	Maximum atom-type E-State: -O-
16	x_{16}	BCUTp-1h	nlow highest polarizability weighted BCUTS
17	x_{17}	MDEO-12	Molecular distance edge between all primary
18	x_{18}	maxsOH	Molecular distance edge between all primary
19	x_{19}	TopoPSA	Topological polar surface area
20	x_{20}	XLogP	XLogP

5 问题二：ERα 生物活性预测模型的建立与求解

5.1 问题二分析

首先，分析第一问得到的 20 个自变量，通过检测发现变量间存在较强的多重共线性，不适合直接线性回归。因此，我们应该选用能够去除多重共线性干扰的线性回归。在此我们分别选用逐步回归、岭回归、LASSO 来进行回归分析。同时建立非线性回归模型，而比较常用的便是多项式回归，通过比较找出更适合本文的回归方法和回归方程。其次，考虑到常规统计模型在稳定性和理论上比较有优势，而机器学习的方法在计算精度上会有不一样的效果，所以我们还应该用机器学习的方法进行预测，同时对线性、非线性回归所

得结果进行验证。因此接下来选择适合本文数据特征，同时对因变量的数据类型没有特定要求的模型，以方便对 IC_{50} 值、pIC_{50} 值这样的连续数据和 Caco-2、CYP3A4、hERG、HOB、MN 这样的分类数据都适用。经过综合考量，我们选取极端梯度提升（XGBoost）和 RF 算法，考虑到 XGBoost 和 RF 算法中有很多难以手动设置的超参数，我们分别在 XGBoost 和 RF 算法的基础上，用智能优化算法中的遗传算法（）GA 进行超参数调优，之后对 GA-XGBoost 和 GA-RF 进行组合，建立基于集成学习方法的 ERα 生物活性组合预测模型。最后选择预测精度最好的模型对给定的 50 个化合物的 IC_{50} 值和 pIC_{50} 值进行预测。

技术路线图如下：

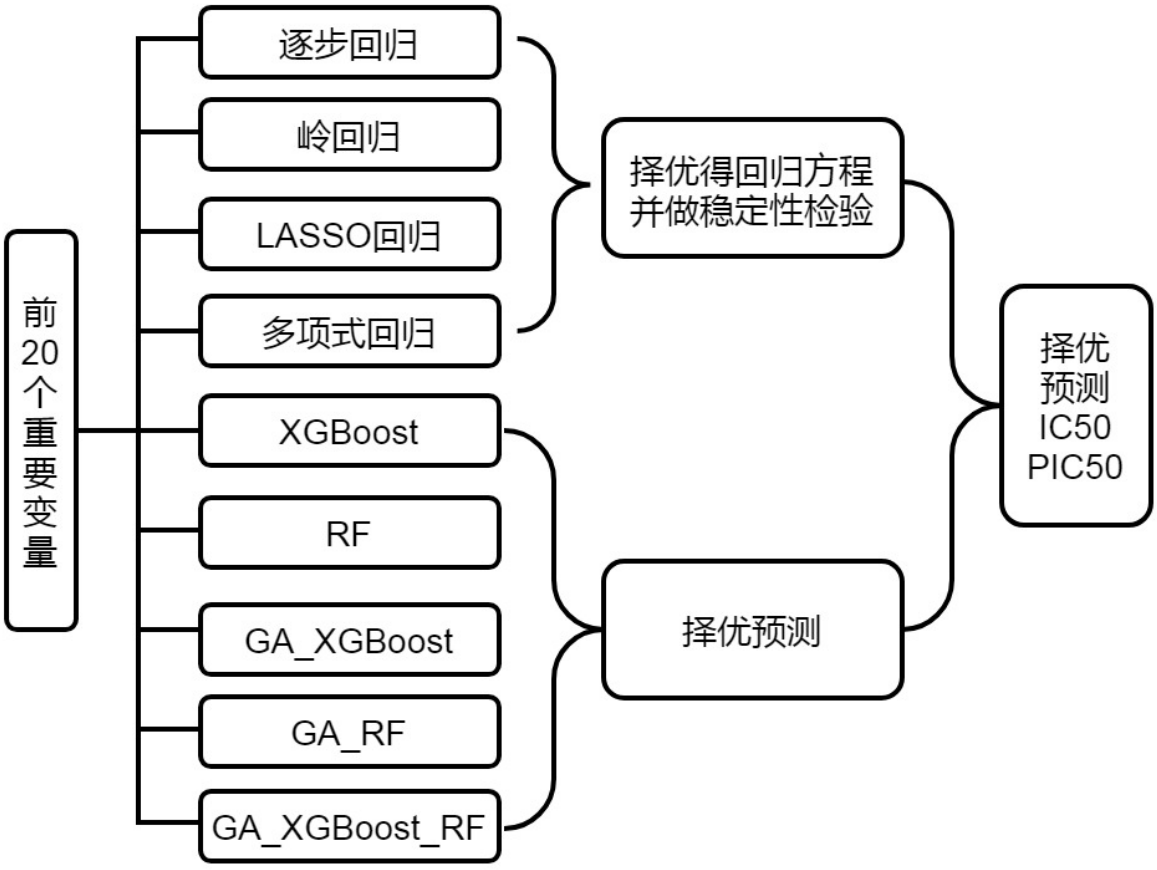


图 5.6　问题二技术路线图

5.2 ERα 生物活性回归预测模型

5.2.1 逐步回归

逐步回归是将偏回归平方和经验显著的自变量逐个的选入模型中，同时每引入一个新变量，需要将原来模型中的其他变量逐个检验，将不显著变量删除，直至不能再引入新变

量，此时回归模型中的所有自变量都是显著的。

5.2.2 岭回归

在线性回归模型中，参数的估计公式为 $\beta = \left(X^T X\right)^{-1} X^T y$, 当 $X^T X$ 接近于奇异时，无法求出 β 值。为了保证参数 β 可求，岭回归模型加入了 L_2 范数的惩罚项，模型为：

$$\|X\beta - y\|^2 + \|\Gamma\theta\|^2$$

其中，我们定义 $\Gamma = \alpha I$，于是：

$$\beta(\alpha) = \left(X^T X + \alpha I\right)^{-1} X^T y$$

其中，I 是单位矩阵。

5.2.3 Lasso 回归

正则化方法是通过加入惩罚项，使回归系数较小的压缩至 0，从而获得具有更高预测准确性和概化能力的模型。Lasso 回归是正则化方法中较为典型的一种方法，它的惩罚项是回归系数的绝对值之和，即：$P^{Lasso}(\beta) = \lambda \sum_{j=1}^{p} |\beta_j|$，那么 Lasso 损失函数的公式可以表示为：

$$L^{Lasso}(\beta) = \|Y - X\beta\|^2 + \lambda W^T \beta$$

5.2.4 多项式回归

多项式回归是研究一个因变量与一个或多个自变量间多项式的回归分析的方法。常见的多项式回归模型有：

一元 m 多项式回归模型：$\hat{y} = b_0 + b_1 x + b_2 x^2 + \cdots + b_m x^m$

二元二次多项式回归模型：$\hat{y} = b_0 + b_1 x_1 + b_2 x_2 + b_3 {x_1}^2 + b_4 {x_2}^2 + b_5 x_1 x_2$

5.2.5 GA_XGBoost_RF 组合预测模型

本文利用 GA 和 XGBoost 算法与 RF 算法进行组合。XGBoost 算法是近年来兴起的一种高效集成学习方法, 已在众多预测领域中取得了应用。XGBoost 算法属于 Boosting 集成学习方法, 而 RF 算法是基于 Bagging 的，将两种不同的集成学习方法进行组合可以结合两种方法的优点。使用 GA 对 XGBoost 进行调参在运行时间和调参效果上优于网格搜索和随机游走 [**?**]。并且 GA 对问题的可行解进行编码, 通过适应度来选择判断基因优劣, 达到对目标函数没有连续和可导的要求, 因此简化了组合模型参数调优和权值调优的复杂程度。

利用 GA 良好的全局搜索能力和灵活性, 对 XGBoost 算法和 RF 算法的参数进行优化, 然后再用其确定 GA_XGBoost 算法和 GA_RF 算法的权值, 最终建立 GA_XGBoost_RF 组

合预测模型。后面将分别介绍参数调优和权值调优的具体内容。GA_XGBoost_RF 算法的流程如图5.7所示。

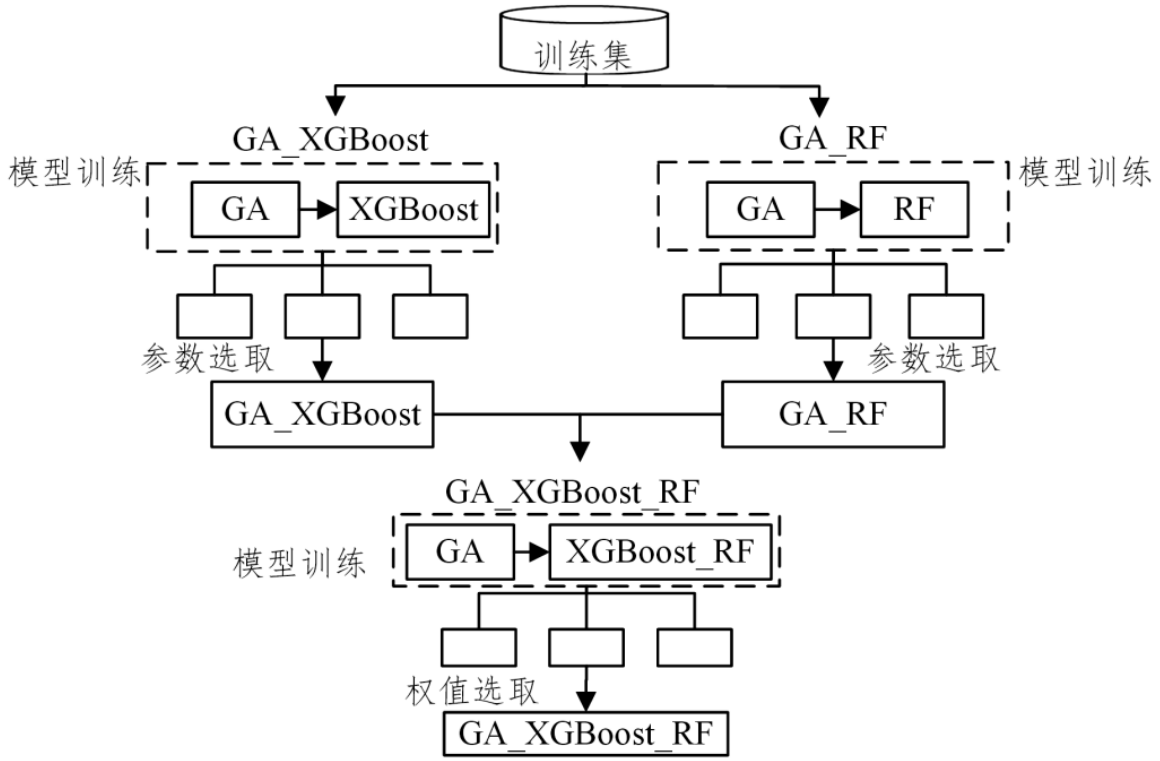


图 5.7　GA_XGBoost_RF 算法流程图

(1)XGBoost 算法

XGBoost 是想将基函数与权重通过 Boosting 思想组合而成的算法，其速度快、通用性强，适合处理大规模数据。对于 n 条 m 维的数据集：

$$\mathrm{D}=\{(\mathrm{x_i},\quad \mathrm{y_i})\}\,(\mathrm{x_i}\in \mathrm{R^m}, \mathrm{y_i}\in \mathrm{R}, \mathrm{i}=1,2,\ldots \mathrm{n})$$

XGBoost 模型表示为：

$$\widehat{y_l}=\sum_{k=1}^{K} f_k\left(x_i\right), f_k\in F(i=1,2,\ldots n)$$

上式中，K 代表树的颗数，x_i 表示第 i 个数据点的特征向量，f_k 表示一颗具体的 CART 决策树，F 是所有 CART 决策树的结构集合。

目标函数包含两部分：

$$Obj=\sum_{i=1}^{n} l\left(y_i,\widehat{y_l}\right)+\sum_{k=1}^{K}\Omega\left(f_k\right)$$

上式中：l 代表训练中存在的误差，是预测值和目标值之间按照特定标准计算的差异程度。Ω 则代表所有 CART 树的复杂度之和。

由于 XGBoost 模型不是一个显式函数，无法用传统的方法进行优化，Xboost 使用加性训练进行优化，也就是每次优化一棵树，直到所有树都得到优化。程序如下：

$$\hat{y}_i^{(0)} = 0$$

$$\widehat{y}_i^{(1)} = \mathrm{f}_1(x_i) = \hat{y}_i^{(0)} + \mathrm{f}_1(x_i)$$

$$\widehat{y}_i^{(2)} = \mathrm{f}_1(x_i) + \mathrm{f}_2(x_i) = \hat{y}_i^{(1)} + \mathrm{f}_2(x_i)$$

$$\cdots\cdots$$

$$\hat{y}_i^{(t)} = \sum_{\mathrm{k}=1}^{\mathrm{t}} \mathrm{f}_{\mathrm{k}}(x_i) = \hat{y}_{\mathrm{i}}^{(\mathrm{t}-1)} + \mathrm{f}_{\mathrm{t}}(x_i)$$

其中，$\hat{y}_i^{(t)}$ 为第 t 次模型的预测值，$\mathrm{f}_{\mathrm{t}}(x_i)$ 为第 t 次加入的新函数。每一轮加入新函数是为了尽可能让目标函数值最大程度的减小。将 $\hat{y}_i^{(t)}$ 带入公式 $Obj = \sum_{i=1}^{n} l(y_i, \widehat{y}_l) + \sum_{k=1}^{K} \Omega(f_k)$ 得到第 t 次模型的目标函数：

$$\mathrm{Obj}^{(t)} = \sum_{i=1}^{n} l(y_i, \widehat{y}_i) + \sum_{i=1}^{t} \Omega(f_i)$$

$$= \sum_{i=1}^{n} l\left(y_{i,} \hat{y}_i^{(t-1)} + f_i(x_i)\right) + \Omega(f_t) + constant$$

采用泰勒展开近似定义误差函数 l，得到目标函数为：

$$\mathrm{Obj}^{(t)} \approx \sum_{i=1}^{n} \left[l\left(y_i, \widehat{y}_i^{(t-1)}\right) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right]$$

$$+\Omega(f_t) + \ constant$$

其中，$g_i = \partial_{\widehat{y}_l(t-1)} l\left(y_{i,}, \widehat{y}_l^{(t-1)}\right)$，$h_i = \partial^2_{\widehat{y}_l^{(t-1)}} l\left(y_i, \hat{y}_l^{(t-1)}\right)$，即误差函数 l 的一阶导数和二阶导数。把不影响优化的常数项移除后，得到最终目标函数：

$$\mathrm{Obj}^{(t)} \approx \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t)$$

(2)RF 算法

随机森林是将 bagging 思想和随机选择特征相结合的一种方法，该方法通过建立很多棵决策树，组成决策树“森林”，再依据多棵树的投票结果进行决策，能够得到一个更加准确和稳定的模型。

随机森林算法的具体过程如下 [**?**]：

1、使用 bootstrap 方法有放回的从原始训练集中随机选取每颗树的训练集，共获取 N 个训练集。

2、构建决策树，假设决策树 t_1 的训练集为样本 1，node 表示 t_1 上的任意一个节点。决策树 t_1 的构建过程：

a）节点 node 随机从样本集 1 中抽取 m 个特征，记为 S_m;

b) 从特征子集 $S_m = S_1, S_2, \cdots S_m$ 中选择最具有分裂能力的特征 $S_k\,(k \in [1, m])$，用 S_k 对节点 node 进行分割。

重复上述 a, b 步骤，直至达到最大树高或满足最小样本数等停止条件。

3、当决策树构建完成后，每个决策树对新输入的样本数据输出一个预测结果，再将其按照统计规则输出最终结果。

随机森林回归预测的组合决策过程：

$$H\left(X\right) = avg\left(\sum h_i\left(x\right)\right)$$

随机森林分类预测的组合决策过程：

$$H\left(x\right) = \arg\max_{V} \sum I\left(h_i\left(x\right) = Y\right)$$

其中，$h_1\left(X\right), h_2\left(X\right), \cdots, h_i\left(X\right)$ 表示若干棵决策树，$H\left(x\right)$ 表示随机森林模型，X 为样本的特征属性，Y 为对应的类别属性，I 为指示函数。

(3)GA 算法

遗传算法（GA）通过模拟自然进化过程来搜索最优解。它是将种群中的每个带有特征的个体看作染色体，然后仿照基因编码产生初代种群，再根据问题域中个体的适应度选择个体，并借助遗传算子进行组合交叉和变异，产生代表新解集的种群。具体步骤如下：

1、初始化种群：随机生成一组染色体；

2、适应值：由适应度函数得到个体在繁衍竞争中的适应值；

3、选择：选出将染色体传给下一代的父母，其他皆淘汰；

4、交叉：以某点为中心，交叉、交换染色体，从而产生新的个体；

5、变异：根据变异概率，对每个新个体进行变异。

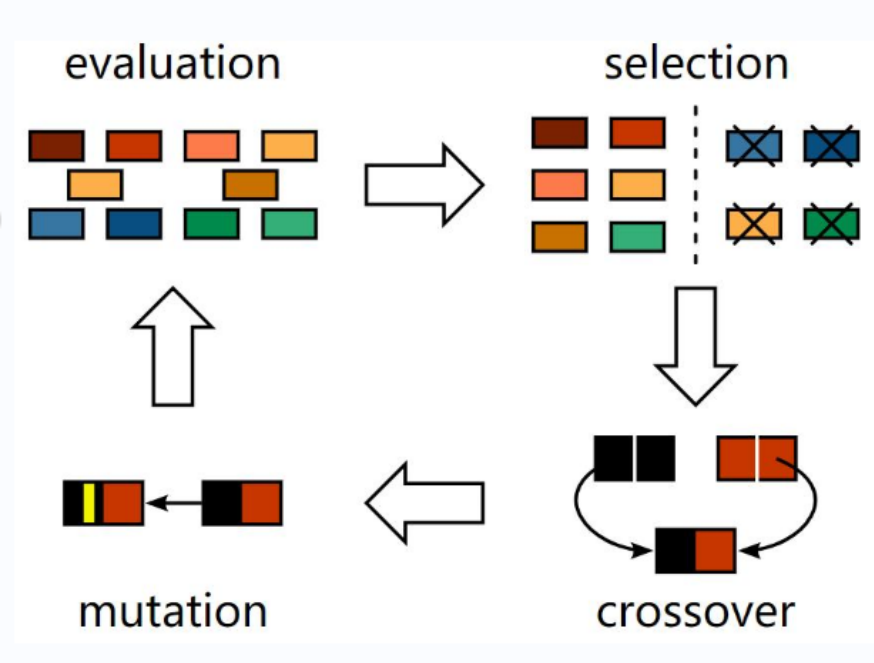


图 5.8　遗传算法演示

遗传算法的伪代码如表5.6:

表 5.6　遗传算法伪代码

遗传算法伪代码
输入：　种群规模 P, 迭代次数 T，交叉概率 PC，变异概率 PM 等参数
1.　遗传代数 t=0
2.　初始化种群 P(t）
3.　计算 P(t）适应度
4.　while（不满足停止准则）do
5.　t=t+1
6.　从 P(t-1）中选择 P(t）
7.　按照 PC 进行交叉
8.　按照 PM 进行变异
9.　产生新的种群 P(t）
10.　计算 P(t）适应度
11.　end while
输出：　最佳个体

(4) 参数优化：GA_XGBoost 和 GA_RF 算法

由于 XGBoost 和 RF 算法中参数多、调节过程繁琐且参数对预测效果有很大影响，所以提出 GA_XGBoost 和 GA_RF 模型，利用 GA 算法的全局搜索、灵活性等优点弥补 XGBoost 和 RF 的缺陷。本文使用交叉验证的平均得分作为适应度函数值，建立 GA_XGBoost 和 GA_RF 模型进行算法参数优化。GA_XGBoost(GA_RF) 算法的伪代码如下：

表 5.7　GA_XGBoost 算法伪代码

GA_XGBoost 伪代码
输入：种群数量 P，迭代次数 T，参数数量 N，优秀个体数量 M
1.　for i←1 to P do
2.　初始化各种群的参数组 $(\theta_{i1}, \theta_{i2}, \ldots, \theta_{iN})$
3.　end for
4.　while（不满足停止准则）do
5.　训练集训练 XGBoost 模型
6.　对验证集进行预测，计算适应度
7.　根据适应度由高到低保留 M 组优秀个体
8.　GA(P,T,N,M)
9.　产生新参数
10.　end while
输出：最佳参数组合

(5) 权值优化：GA_XGBoost_RF 算法

对调参后的 GA_XGBoost 模型和 GA_RF 模型建立变权组合预测模型, 其中最重要的是两个模型各自权值的确定, 本文利用 GA 来确定两个模型的权值。

首先利用参数优化中得到的参数组合在训练集上进行训练, 构建 GA_XGBoost 和 GA_RF 模型。其次使用 GA 优化权值, 设置权值参数的范围、遗传算法的迭代次数、初始种群的数量, 然后随机生成 P 组初始值, 若不满足停止条件, 则将种群中的每个个体作为组合模型的权值对训练集进行预测, 以真实值与预测值的均方误差为适应度函数, 以权值之和等于 1 为约束条件, 从种群中选取 u 个优秀个体。

$$fit = \sqrt{\frac{1}{m}(w_1\hat{y}_{XGB} + w_2\hat{y}_{RF} - y_{True})^2}$$

$$w_1 + w_2 = 1$$

其中,m 为训练集中的样本个数,w_1 和 w_2 分别为 XGBoost 和 RF 模型的权值,$\hat{y}_{XGB}$ 和 $\hat{y}_{RF}$ 分别为 XGBoost 和 RF 模型对训练集的预测值,y_{True} 为训练集的真实值。

对选取的优秀个体进行交叉、变异, 从而产生新的个体, 循环这个过程直到满足条件时停止, 从历代种群中选择最优值作为最终结果, 得到组合模型的权值组合, 建立 GA_XGBoost_RF 模型。GA_XGBoost_RF 算法如下：

GA_XGBoost_RF 算法

输入: 人口规模 P、迭代次数 T、杰出个人数量 U

输出: 最佳权重

1. 训练 $GA\_XGBoost$ 和 GA_RF 模型
2. 初始化权重
3. while 不满足终止条件时
4. 预测训练集
5. 计算适应度
6. 根据适应度选出 u 组最好权重组合
7. 进行遗传、变异等运算
8. 生成新的权重组合
9. end while

5.3 模型评价

本文主要采用以下几种方法对模型进行评价：

1、均方根误差：

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2},$$

该值越大，模型误差越大。

2、平均绝对误差：

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i|,$$

该值越大，模型误差越大。

3、平均绝对百分比误差：

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

该值越大，模型误差越大。但是需要注意的是，当 y_i 有 0 值时，不能使用。

5.4 模型求解

首先我们分析第一问得到的 20 个自变量，通过 SPSS 进行线性回归并检验其共线性，结果如表5.8：

表 5.8　共线性统计

	未标准化系数		标准化	t	显著性	β 的 95.0% 置信区间		共线性统计	
	β	标准错误	系数 β			下限	上限	容差	VIF
(常量)	0.265	0.04		6.577	0	0.186	0.344		
MDEC-23	0.386	0.05	0.374	7.659	0	0.287	0.485	0.098	10.19
BCUTc-1l	0.173	0.037	0.117	4.622	0	0.099	0.246	0.364	2.74
maxHsOH	0.144	0.157	0.222	0.917	0.359	-0.164	0.452	0.004	249.9
minHsOH	-0.102	0.156	-0.15	-0.65	0.516	-0.408	0.205	0.004	227.4
minsOH	0.475	0.434	0.911	1.095	0.274	-0.376	1.325	0	2955.
LipoaffinityIndex	0.299	0.094	0.261	3.17	0.002	0.114	0.485	0.035	28.94
ATSp5	-0.417	0.063	-0.426	-6.582	0	-0.541	-0.293	0.056	17.90
nC	1.175	0.187	1.201	6.28	0	0.808	1.542	0.006	156.0
SsOH	-0.228	0.153	-0.219	-1.495	0.135	-0.528	0.071	0.011	91.40
BCUTc-1h	-0.215	0.023	-0.217	-9.231	0	-0.26	-0.169	0.422	2.36
MLogP	-1.289	0.149	-1.132	-8.646	0	-1.581	-0.997	0.014	73.16
SHsOH	0.137	0.127	0.163	1.084	0.279	-0.111	0.385	0.01	96.35
MLFER_A	0.307	0.053	0.222	5.792	0	0.203	0.411	0.159	6.28
WTPT-5	-0.264	0.04	-0.237	-6.648	0	-0.342	-0.186	0.185	5.41
maxssO	0.02	0.011	0.05	1.901	0.057	-0.001	0.042	0.339	2.94
BCUTp-1h	0.342	0.037	0.353	9.172	0	0.269	0.415	0.158	6.31
MDEO-12	-0.12	0.031	-0.123	-3.9	0	-0.18	-0.059	0.237	4.22
maxsOH	-0.418	0.463	-0.767	-0.902	0.367	-1.325	0.49	0	3084.5
TopoPSA	-0.166	0.052	-0.146	-3.179	0.002	-0.269	-0.064	0.11	9.05
XLogP	-0.116	0.052	-0.061	-2.228	0.026	-0.218	-0.014	0.31	3.22

由表可知，多个变量的 p 值均大于 0.05，maxHsOH 的 p 值为 0.359，minHsOH 的 p 值为 0.516，明显大于 0.05，也就是说纳入线性回归模型是没有统计学意义的；同时，多个变量的 VIF 都很大，如 maxsOH 的 VIF 值为 3084.589，minsOH 的 VIF 值为 2955.090。结合两者的含义，我们可以推断出变量间存在较强的多重共线性，不适合直接线性回归。且

线性回归调整后的 R^2 为 0.561，效果不甚满意，综上，我们选用能够去除多重共线性的线性回归，如逐步回归、岭回归和 LASSO 回归。

求解逐步回归得表**??**：

由表可知，逐步回归剔除了 maxHsOH、minsOH、SsOH、SHsOH、maxssO、maxsOH 这 6 个变量，保留了 14 个变量，得出的回归方程为：

但是逐步回归调整后的 R^2 为 0.561，与直接线性回归并没有改变，这仍需要我们进一步推断。同样地，求出岭回归调整后的 R^2 为 0.573，LASSO 调整后的 R^2 为 0.502，都没有明显提升，这便提示我们应该考虑用非线性回归。因此，我们用最常见的多项式回归，并且考虑到变量个数太多，不方便考虑交叉项，最终选用不带交叉项的二次多项式回归。

求解二次多项式回归得表5.9：

表 5.9　逐步回归结果

	未标准化系数		标准化	t	显著性	β 的 95.0% 置信区间	
	β	标准错误	系数 β			下限	上限
(常量)	0.275	0.038		7.222	0.000	0.201	0.350
MDEC-23	0.376	0.048	0.365	7.758	0.000	0.281	0.471
BCUTc-1h	-0.216	0.020	-0.218	-10.642	0.000	-0.256	-0.176
BCUTp-1h	0.360	0.036	0.372	10.126	0.000	0.291	0.430
ATSp5	-0.441	0.060	-0.451	-7.401	0.000	-0.558	-0.324
LipoaffinityIndex	0.349	0.090	0.304	3.874	0.000	0.172	0.525
MLFER_A	0.249	0.034	0.180	7.292	0.000	0.182	0.316
MLogP	-1.299	0.144	-1.141	-8.993	0.000	-1.583	-1.016
WTPT-5	-0.265	0.038	-0.238	-7.004	0.000	-0.340	-0.191
nC	1.197	0.176	1.223	6.794	0.000	0.851	1.543
BCUTc-1l	0.153	0.033	0.104	4.610	0.000	0.088	0.219
TopoPSA	-0.170	0.050	-0.149	-3.391	0.001	-0.268	-0.072
minHsOH	0.130	0.017	0.192	7.427	0.000	0.096	0.164
MDEO-12	-0.090	0.027	-0.092	-3.384	0.001	-0.142	-0.038
XLogP	-0.144	0.051	-0.076	-2.829	0.005	-0.243	-0.044

由表可知，逐步回归剔除了 maxHsOH、minsOH、SsOH、SHsOH、maxssO、maxsOH 这 6 个变量，保留了 14 个变量，得出的回归方程为：

但是逐步回归调整后的 R^2 为 0.561，与直接线性回归并没有改变，这仍需要我们进一

步推断。同样地，求出岭回归调整后的 R^2 为 0.573，LASSO 调整后的 R^2 为 0.502，都没有明显提升，这便提示我们应该考虑用非线性回归。因此，我们用最常见的多项式回归，并且考虑到变量个数太多，不方便考虑交叉项，最终选用不带交叉项的二次多项式回归。

求解二次多项式回归得表5.10：

表 5.10　多项式回归结果

	未标准化系数		标准化	t	显著性	β 的 95.0% 置信区间	
	β	标准错误	系数 β			上限	下限
(常量)	1.351	0.287		4.702	0	0.788	1.915
MDEC-23	-0.289	0.146	-0.28	-1.979	0.048	-0.575	-0.003
BCUTc-1l	-0.957	0.315	-0.651	-3.041	0.002	-1.574	-0.34
maxHsOH	0.521	0.332	0.801	1.567	0.117	-0.131	1.172
minHsOH	-1.22	0.326	-1.801	-3.743	0	-1.858	-0.581
LipoaffinityIndex	0.363	0.193	0.316	1.881	0.06	-0.015	0.741
ATSp5	1.216	0.176	1.243	6.901	0	0.87	1.561
nC	-0.262	0.401	-0.267	-0.653	0.514	-1.047	0.524
SsOH	-2.35	0.567	-2.252	-4.146	0	-3.462	-1.238
BCUTc-1h	-0.37	0.073	-0.374	-5.075	0	-0.513	-0.227
MLogP	-0.337	0.454	-0.296	-0.741	0.459	-1.227	0.554
SHsOH	2.668	0.507	3.167	5.265	0	1.674	3.661
MLFER_A	0.252	0.094	0.182	2.692	0.007	0.068	0.436
WTPT-5	-0.317	0.064	-0.284	-4.926	0	-0.443	-0.191
maxssO	-0.518	0.075	-1.267	-6.931	0	-0.665	-0.372
BCUTp-1h	-1.801	0.335	-1.857	-5.383	0	-2.457	-1.145
MDEO-12	-0.095	0.057	-0.098	-1.663	0.097	-0.208	0.017
maxsOH	0.933	0.728	1.712	1.282	0.2	-0.494	2.36
TopoPSA	0.053	0.109	0.046	0.485	0.628	-0.161	0.266
XLogP	0.214	0.095	0.113	2.24	0.025	0.027	0.4
S-MDEC-23	0.63	0.141	0.586	4.465	0	0.354	0.907
S-BCUTc-1l	-0.997	0.299	-0.723	-3.335	0.001	-1.583	-0.411
S-maxHsOH	-0.577	0.229	-0.674	-2.518	0.012	-1.027	-0.128
S-minHsOH	0.41	0.17	0.486	2.406	0.016	0.076	0.744
S-minsOH	0.516	0.28	0.856	1.846	0.065	-0.032	1.065
S-LipoaffinityIndex	-0.153	0.226	-0.103	-0.679	0.497	-0.596	0.289
S-ATSp5	-1.485	0.156	-1.353	-9.529	0	-1.791	-1.179
S-nC	0.948	0.361	0.883	2.623	0.009	0.239	1.657
S-SsOH	0.904	0.368	0.484	2.458	0.014	0.183	1.625
S-BCUTc-1h	0.273	0.085	0.225	3.226	0.001	0.107	0.439
S-MLogP	-0.422	0.499	-0.344	-0.845	0.398	-1.4	0.557
S-SHsOH	-1.388	0.288	-1.114	-4.817	0	-1.954	-0.823
S-MLFER_A	-0.141	0.1	-0.08	-1.413	0.158	-0.337	0.055
S-WTPT-5	0.147	0.074	0.08	1.981	0.048	0.001	0.293
S-maxssO	0.603	0.085	1.334	7.117	0	0.437	0.769

二次多项式回归调整后的 R^2 为 0.616，标准误差为 0.112，将几种方法做一下简单比较，就会发现二次多项式回归效果会更好些。而得到的二次多项式回归方程为：

$$Y = 1.124 - 0.388T_{660} + 0.491T_{477} - 1.605T_{358} + 0.954T_{411} + 1.014T_{36} - 0.554T_{40} - 2.29T_{292}$$

$$-0.518T_{41} + 3.316T_{239} + 0.104T_{674} - 0.102T_{725} - 1.591T_{532} - 1.735T_{43} + 0.181T_{728}$$

$$+0.62T_{660}^2 - 0.612T_{40}^2 - 0.578T_{477}^2 + 0.564T_{358}^2 - 1.204T_{36}^2 + 0.257T_{12}^2 + 0.429T_{292}^2 + 0.407T_{41}^2$$

$$-1.199T_{239}^2 + 0.025T_{725}^2 + 1.625T_{532}^2 + 1.904T_{43}^2 - 0.223T_{728}^2$$

为了最后与分类预测的变量符号定义保持一致，我们将在分类预测方程得出后统一定义符号。

由组合预测模型 $GA_XGBoost_GF$, 对测试集中的 50 个化合物预测的部分结果如图**??**

表 5.11　IC50、pIC50 部分预测结果

SMILES	IC50_nM	pIC50
1	17.9795	7.7452
2	11.7993	7.9281
3	16.7308	7.7765
4	15.9969	7.796
5	37.9719	7.4205
6	24.6883	7.6075
7	25.3371	7.5962
8	22.1709	7.6542
9	17.2616	7.7629
10	19.8357	7.7026

6 问题三：化合物 ADMET 性质分类预测模型的建立与求解

6.1 问题三分析

根据题目要求，本文用两种方法分别建立分类预测模型。首先利用组合变量选择法筛选出变量，再用逻辑回归分别求出针对化合物的 Caco-2、CYP3A4、hERG、HOB、MN 的

逻辑回归模型。接着又利用 XGBoost、RF 分别筛选出合适的变量，分别将这些变量用于与 XGBoost、RF、GA_XGBoost、GA_RF 以及 GA_XGBoost_GF 预测模型中，择优选出最适合的预测模型，最终预测出 test 表中化合物 ADMET 性质。

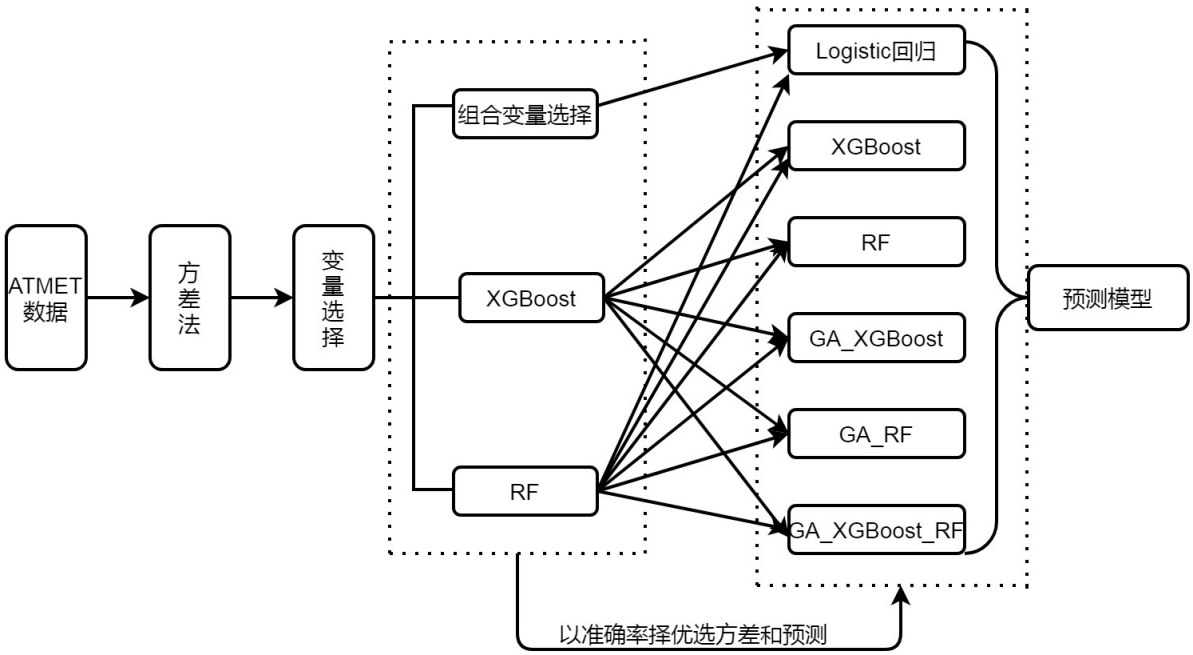


图 6.9　问题三技术路线图

6.2 化合物 ADMET 性质分类预测模型

6.2.1 逻辑回归

逻辑回归模型可以看成一个被 logistic 方程归一化后的线性回归模型。其公式为：

$$P\left(y=1|\,x;\theta\right)=\frac{1}{1+e^{-\theta^{T}x}}$$

其中 $P\left(y=1|\,x;\theta\right)$ 表示：在给定 x 条件下事件 y 发生的条件概率，而 θ 是条件概率的参数。

逻辑回归的代价函数为：

$$J\left(\theta\right)=\frac{1}{m}\sum_{i=1}^{m}y^{(i)}\log\left(h_{\theta}\left(x^{(i)}\right)\right)+\left(1-y^{(i)}\right)\log\left(1-h_{\theta}\left(x^{(i)}\right)\right)$$

为了解决过拟合问题，需要再进行正则化，正则化之后得到：

$$J\left(\theta\right)=\frac{1}{m}\sum_{i=1}^{m}y^{(i)}\log\left(h_{\theta}\left(x^{(i)}\right)\right)+\left(1-y^{(i)}\right)\log\left(1-h_{\theta}\left(x^{(i)}\right)\right)+\frac{\lambda}{2m}\sum_{j=1}^{n}{\theta_{j}}^{2}$$

6.2.2 组合预测模型

在问题二中已经详细描述了组合预测模型，在这里就不在赘述。但是需要强调的是，在使用组合预测模型分别对化合物的 5 种性质做分类预测模型时，需要利用 GA 分别对 XGBoost、RF 进行调参，得到的具体参数如表 6.12、6.13所示：

表 6.12　GA_XGBoost 最优参数

	Caco-2	CYP3A4	hERG	HOB	MN
n_estimators	140	160	70	90	140
eta	0.1816	0.1939	0.2	0.1571	0.1877
max_depth	8	7	6	8	3
min_child_weight	2	1	2	7	2

表 6.13　GA_RF 最优参数

	Caco-2	CYP3A4	hERG	HOB	MN
n_estimators	120	110	100	140	60
max_depth	16	12	15	16	12

6.3 模型评价

1、霍斯默 -莱梅肖检验：用于检验逻辑回归模型的拟合优度，当该建议结果大于 0.05 时，说明该模型拟合程度较好。

2、准确率：预测正确的样本占全部样本的比例，公式为：

$$accuracy(y,\hat{y}) = \frac{1}{n}\sum_{i=1}^{n} 1\left(\hat{y}_i = y_i\right)$$

3、召回率：针对原始样本而言的指标，表示真正样本中有多少预测对了，公式为：

$$\frac{TP}{TP+FN}$$

4、F1 分数 (F1-score) 是 precision 和 recall 的调和平均数（倒数平均数），将它们看成同等重要。通常 precision 和 recall 不能兼得，查准率高了查全率可能会偏低，查全率高了查准率可能会偏低，使用 F1 分数可以综合考虑它们二者。

$$F_1 = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} = \frac{2 \times precision \times recall}{precision + recall}$$

5、ROC 曲线, 绘制 ROC 曲线需要计算真正例率（TPR) 和假正例率（FPR), 其定义为：$TPR = \frac{TP}{TP+FN}$，$FPR = \frac{FP}{TN+FP}$。ROC 曲线以图形方式组合 TPR 和 FPR，可以直观反映分类器的真正例率和假正利率的关系。但是在实际应用中 ROC 曲线可能出现交叉情况，无法直观判断孰好孰坏，所以通常依据 ROC 曲线下方的面积 AUC 来判断，AUC 越靠近 1，说明模型性能越好。

6.4 模型求解

对于第一个性质，首先在 spss 软件中输入变量：WPATH, ECCEN, MLFER_L, ETA_Beta_s, WTPT1, MW, ETA_Beta, MLFER_E, sumI, TopoPSA, ATSm2, MLFER_S, ATSm1, ETA_Eta_F_L, minHBa, ETA_BetaP_ns_d, Kier3, ETA_Shape_P, ATSm3, MDEC22 得到结果如表 6.14

表 6.14　基于性质 1 的第一次逻辑回归结果

	β	标准误差	瓦尔德	自由度	显著性	Exp(β)
WPATH	-0.005	0.001	37.461	1	0.000	0.995
ECCEN	0.007	0.003	7.753	1	0.005	1.007
MLFER_L	0.064	0.141	0.202	1	0.653	1.066
ETA_Beta_s	-0.162	0.198	0.667	1	0.414	0.850
WTPT1	0.736	0.130	32.176	1	0.000	2.088
MW	-0.077	0.022	11.786	1	0.001	0.926
ETA_Beta	-0.612	0.117	27.302	1	0.000	0.542
MLFER_E	-0.328	0.393	0.696	1	0.404	0.720
sumI	-0.041	0.023	3.286	1	0.070	0.960
TopoPSA	-0.038	0.006	37.894	1	0.000	0.963
ATSm2	0.621	0.097	40.800	1	0.000	1.861
MLFER_S	-0.068	0.322	0.044	1	0.834	0.935
ATSm1	0.033	0.019	2.983	1	0.084	1.034
ETA_Eta_F_L	1.815	0.414	19.245	1	0.000	6.139
minHBa	-0.138	0.035	15.710	1	0.000	0.871
ETA_BetaP_ns_d	0.809	4.380	0.034	1	0.853	2.245
Kier3	-0.183	0.155	1.393	1	0.238	0.833
ETA_Shape_P	26.179	3.270	64.105	1	0.000	233992485813.381
ATSm3	-0.233	0.060	15.359	1	0.000	0.792
MDEC22	-0.026	0.021	1.506	1	0.220	0.974
常量	-2.213	1.637	1.827	1	0.176	0.109

第一次通过 spss 求解得出霍斯默 -莱梅肖检验 P 值为 0.0836，大于 0.05，说明拟合优度很好。但是有很多变量的系数不显著，需要去掉这些不显著的变量，鉴于常量很重要，先保留看再跑一次的效果，结果见表 6.15：

表 6.15　基于性质 1 的第二次逻辑回归结果

	β	标准误差	瓦尔德	自由度	显著性	Exp(β)
WPATH	-0.005	0.001	45.995	1	0.000	0.995
ECCEN	0.008	0.002	11.575	1	0.001	1.008
WTPT1	0.505	0.049	105.107	1	0.000	1.656
MW	-0.054	0.009	34.705	1	0.000	0.947
ETA_Beta	-0.466	0.094	24.336	1	0.000	0.628
TopoPSA	-0.043	0.005	89.101	1	0.000	0.958
ATSm2	0.535	0.081	43.732	1	0.000	1.708
ETA_Eta_F_L	1.171	0.329	12.685	1	0.000	3.224
minHBa	-0.126	0.028	19.948	1	0.000	0.881
ETA_Shape_P	23.702	2.519	88.502	1	0.000	19661334271.984
ATSm3	-0.181	0.037	23.752	1	0.000	0.835
常量	-2.880	1.555	3.430	1	0.064	0.056

这次得出的霍斯默 -莱梅肖检验依然大于 0.05，说明拟合优度很好。准确率从之前的 85.4 升至 85.5，虽然只有一点提升，但是常量的 P 值降低了很多，几乎接近 0.05，并且减少了多个变量，综合以上分析说明此次操作是非常有效的，最新的这个模型可以保留。那么经 Logistic 回归得到的表达式为：

$$P=\frac{-2.880(0.995)^{T_{726}}(1.008)^{T_{106}}(1.656)^{T_{721}}(0.947)^{T_{720}}(0.628)^{T_{614}}(0.958)^{T_{717}}(1.708)^{T_{28}}(3.224)^{T_{632}}(0.881)^{T}}{-2.880(0.995)^{T_{726}}(1.008)^{T_{106}}(1.656)^{T_{721}}(0.947)^{T_{720}}(0.628)^{T_{614}}(0.958)^{T_{717}}(1.708)^{T_{28}}(3.224)^{T_{632}}(0.881)^{T_{347}}}$$

对于第二个性质，首先输入的变量：SP4, SP3, ETA_Eta_L, VP1, VP2, VP0, VP3, apol, ETA_Alpha, Zagreb, ETA_Eta_R, ETA_Eta_R_L, ETA_Eta, SP7, ETA_Beta_s, minHBa, ETA_dEpsilon_D, ATSm1, ATSm3, WPATH，得到结果如表 6.16所示：

表 6.16　基于性质 2 的逻辑回归第一次结果

	β	标准误差	瓦尔德	自由度	显著性	Exp(β)
SP4	1.575	0.443	12.617	1	0.000	4.832
SP3	-2.771	0.647	18.328	1	0.000	0.063
ETA_Eta_L	5.076	0.934	29.514	1	0.000	160.186
VP1	-7.918	1.226	41.694	1	0.000	0.000
VP2	-0.477	0.682	0.489	1	0.484	0.621
VP0	3.797	0.713	28.380	1	0.000	44.581
VP3	3.969	0.679	34.157	1	0.000	52.911
apol	-0.298	0.116	6.623	1	0.010	0.742
ETA_Alpha	-0.199	1.218	0.027	1	0.870	0.820
Zagreb	0.139	0.100	1.912	1	0.167	1.149
ETA_Eta_R	-0.103	0.142	0.523	1	0.469	0.902
ETA_Eta_R_L	1.442	1.320	1.193	1	0.275	4.229
ETA_Eta	-0.853	0.295	8.384	1	0.004	0.426
SP7	0.198	0.420	0.222	1	0.637	1.219
ETA_Beta_s	-0.126	0.381	0.109	1	0.741	0.882
minHBa	-0.280	0.046	36.294	1	0.000	0.756
ETA_dEpsilon_D	-39.293	6.773	33.653	1	0.000	0.000
ATSm1	-0.013	0.019	0.436	1	0.509	0.987
ATSm3	-0.048	0.070	0.466	1	0.495	0.953
WPATH	0.002	0.001	4.637	1	0.031	1.002
常量	-9.249	3.551	6.785	1	0.009	0.000

由表6.16可知有很多变量的系数不显著，且霍斯默 -莱梅肖检验小于 0.05，说明拟合优度不好，将不显著变量删除，但是保留常数项，重新跑一次，结果见表 6.17:

表 6.17　基于性质 2 的第二次逻辑回归结果

	β	标准误差	瓦尔德	自由度	显著性	Exp(β)
SP4	2.376	0.318	55.662	1	0.000	10.762
SP3	-1.588	0.400	15.775	1	0.000	0.204
ETA_Eta_L	5.168	0.710	52.958	1	0.000	175.618
VP1	-7.067	0.975	52.577	1	0.000	0.001
VP0	2.603	0.440	34.922	1	0.000	13.505
VP3	2.851	0.457	38.985	1	0.000	17.305
apol	-0.101	0.056	3.248	1	0.072	0.904
ETA_Eta	-0.878	0.178	24.437	1	0.000	0.416
minHBa	-0.277	0.036	59.518	1	0.000	0.758
ETA_dEpsilon_D	-40.969	6.256	42.893	1	0.000	0.000
WPATH	0.004	0.001	37.887	1	0.000	1.004
常量	-3.098	2.264	1.874	1	0.171	0.045

此次霍斯默 -莱梅肖检验大于 0.05，说明拟合优度很好，同时准确率从之前的 91.7 升至 92.2，因此此次操作是有很大进展的。但是表中除常量外依然有元素 P 值大于 0.05，所以我们继续删除该变量从新计算，结果见表6.18：

表 6.18　基于性质 2 的第三次逻辑回归结果

	B	标准误差	瓦尔德	自由度	显著性	Exp(B)
SP4	2.424	0.315	59.123	1	0.000	11.296
SP3	-1.702	0.394	18.638	1	0.000	0.182
ETA_Eta_L	5.260	0.701	56.260	1	0.000	192.389
VP1	-7.893	0.867	82.907	1	0.000	0.000
VP0	2.758	0.433	40.614	1	0.000	15.773
VP3	3.247	0.405	64.355	1	0.000	25.708
ETA_Eta	-0.960	0.170	31.981	1	0.000	0.383
minHBa	-0.295	0.034	73.770	1	0.000	0.744
ETA_dEpsilon_D	-39.466	6.110	41.727	1	0.000	0.000
WPATH	0.004	0.001	37.313	1	0.000	1.004
常量	-3.653	2.229	2.686	1	0.101	0.026

这次得出的霍斯默 -莱梅肖检验依然大于 0.05，说明拟合优度很好。准确率从之前的 92.2 降至 91.9，稍微有点降低，但是常量的 P 值降低了很多，并且减少了一个变量，综合以上分析说明此次操作还是成功的，最新的这个模型可以保留。那么经 Logistic 回归得到的表达式为：

$$P=\frac{0.026(11.296)^{T_{92}}(0.182)^{T_{91}}(192.389)^{T_{629}}(0.947)^{T_{720}}(0.628)^{T_{614}}(0.000)^{T_{97}}(15.773)^{T_{96}}(25.708)^{T_{99}}(0.383)^{T_{62}}}{0.026(11.296)^{T_{92}}(0.182)^{T_{91}}(192.389)^{T_{629}}(0.947)^{T_{720}}(0.628)^{T_{614}}(0.000)^{T_{97}}(15.773)^{T_{96}}(25.708)^{T_{99}}(0.383)^{T_{62}}}$$

依此类方法得到，性质 3 的逻辑回归模型为：

$$P=\frac{0.659(1.249)^{T_{54}}(1.009)^{T_{106}}(0.998)^{T_{726}}(2.187)^{T_{648}}(1.809)^{T_{588}}(0.012)^{T_{586}}(1553.125)^{T_{40}}(16.79)^{T_{464}}(0.651)^{T_{64}}}{0.659(1.249)^{T_{54}}(1.009)^{T_{106}}(0.998)^{T_{726}}(2.187)^{T_{648}}(1.809)^{T_{588}}(0.012)^{T_{586}}(1553.125)^{T_{40}}(16.79)^{T_{464}}(0.651)^{T_{64}}}$$

性质 4 的逻辑回归模型为：

$$P=\frac{1.265(0.107)^{T_{226}}(1.291)^{T_{42}}(1.142)^{T_{725}}(1.136)^{T_{662}}(0.520)^{T_{99}}(1.056)^{T_{293}}(16.026)^{T_{586}}(1.066)^{T_{95}}}{1.265(0.107)^{T_{226}}(1.291)^{T_{42}}(1.142)^{T_{725}}(1.136)^{T_{662}}(0.520)^{T_{99}}(1.056)^{T_{293}}(16.026)^{T_{586}}(1.066)^{T_{95}}+1}$$

性质 5 的逻辑回归模型为：

$$P = \frac{(1.869)^{T_{725}}(0.854)^{T_{723}}(1.602E+18)^{T_{604}}(0.339)^{T_{393}}(3.330)^{T_{519}}(3.007)^{T_{643}}(0.323)^{T_{491}}(0.680)^{T_{632}}(2.419E+}{(1.869)^{T_{725}}(0.854)^{T_{723}}(1.602E+18)^{T_{604}}(0.339)^{T_{393}}(3.330)^{T_{519}}(3.007)^{T_{643}}(0.323)^{T_{491}}(0.680)^{T_{632}}(2.419E+}$$

再利用 XGBoost、RF、GA_XGBoost、GA_RF、GA_XGBoost_RF 对化合物的 5 种性质求解分类预测模型并进行预测。在求解过程中，我们分别对每种性质依次使用这 5 个模型求解，然后选取效果最好的模型。

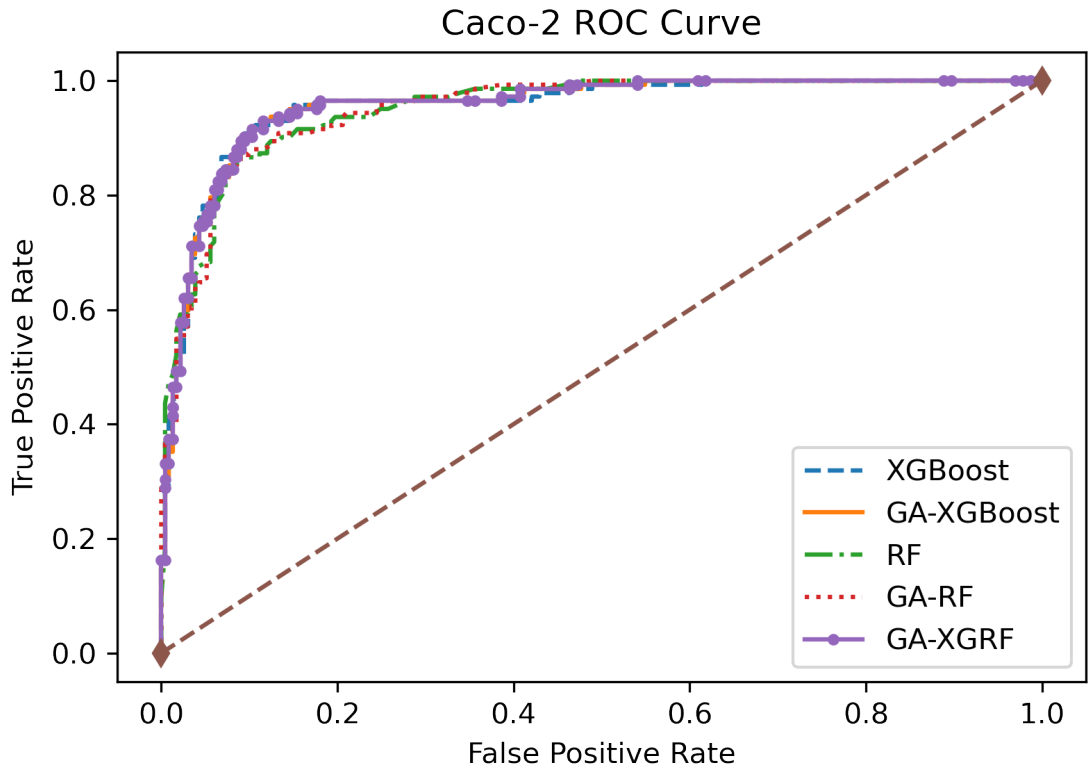


图 6.10　Caco-2 ROC 曲线

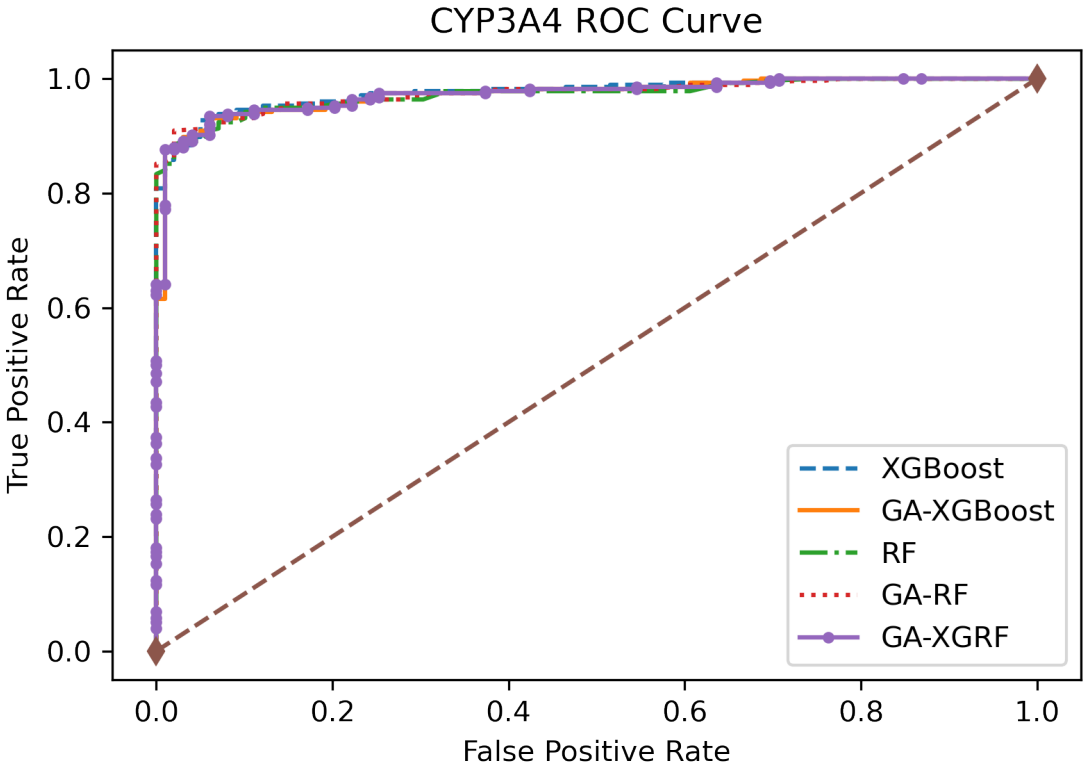


图 6.11　CYP3A4 ROC 曲线

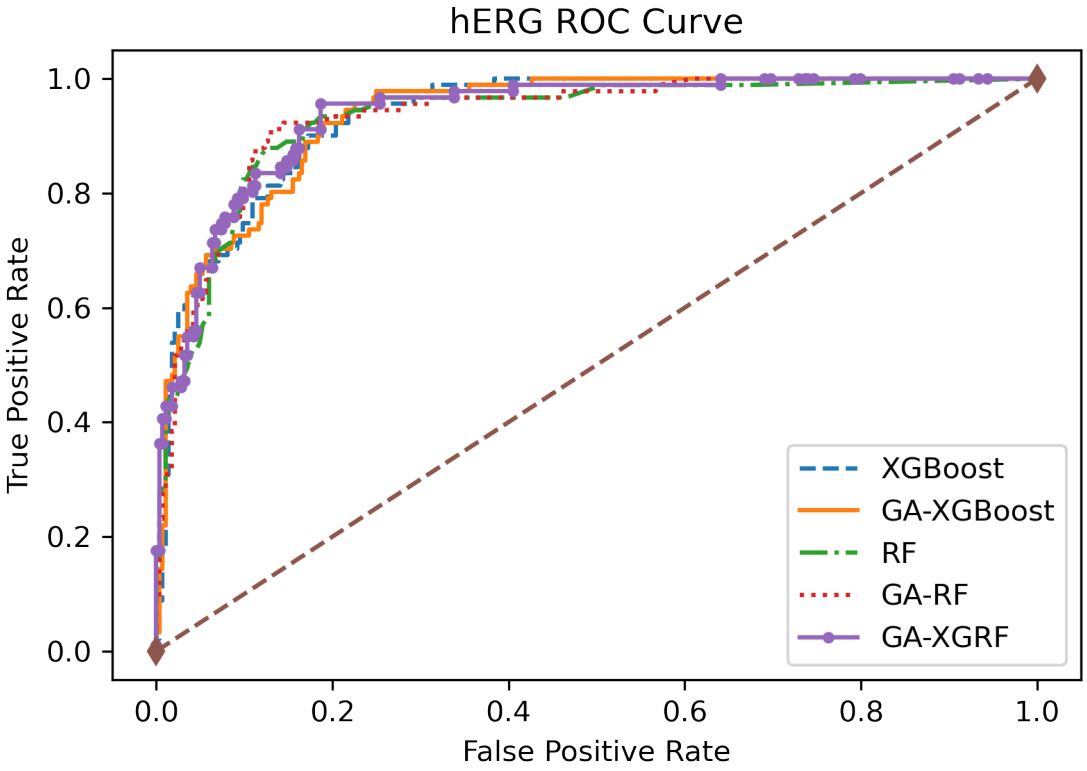


图 6.12　hERGROC 曲线

其中，对于 Caco-2、CYP3A4 和 MN 性质, 我们选择了 GA_XGBoost 模型；hERG 和 HOB 性质，我们选择了 GA_RF 模型。最终预测的部分结果如表**??**所示：

表 6.19　化合物 ADMET 性质预测部分结果

SMILES	Caco-2	CYP3A4	hERG	HOB	MN
1	1	1	1	1	0
2	1	1	1	1	0
3	1	1	1	1	0
4	1	1	1	1	0
5	1	1	1	1	0
6	1	1	1	1	0
7	1	1	1	0	0
8	1	1	1	1	0
9	1	1	1	1	0
10	1	1	1	0	1

7 问题四：ERα 生物活性多目标优化模型

在此我们需要建立一个多目标优化模型，目标一，也是最主要的目标是使化合物能够对抑制 ERα 具备更好的生物活性，目标二是其有更好的 ADMET 性质。同时还有隐含的约束条件，即根据变量的实际含义，找出其常规以及约定俗成的限制。比如，表示原子个数的变量应该是非负整数，而同一原子的最小原子类型 E-State 应该小于等于最大原子类型 E-State。

7.1 基于预测模型的多目标优化模型的建立

在建立优化模型之前，我们首先需要将之前得出的回归方程及变量做整理。鉴于涉及 6 个回归方程，54 个自变量，符号若不统一难以理清，我们将这 54 个变量做统一标记，将其按照在 20 个变量时的排列顺序再按序排列，分别记为 x_{ij}，而因变量分别记为 $y_0, y_1, ..., y_5$，具体命名见前文的符号说明。

结合问题实际情况，忽略一些影响非常小的因素，然后我们构建两个目标函数，ERα 的生物活性最大和具备较好 ADMET 性质的个数最大。

(1) 规划目标一：

依据问题二所得的 ERα 的生物活性多项式回归方程，构建此目标函数。

用连加符号可以将其简写为：

$$y_0 = \alpha_0 + \sum_{j=1}^{m_0} \alpha_j x_{0j} + \sum_{j=m_0+1}^{2m_0} \alpha_j x_{0j}^2$$

(2) 规划目标二：

依据问题三所得的 5 个 ADMET 性质 Logistic 回归方程，构建此目标函数。

用连乘符号可以将其简写为：

$$\begin{aligned} y_i &= \frac{\beta_{i0}\beta_{i1}^{x_{i1}}\beta_{i2}^{x_{i2}}\cdots\beta_{im_i}^{x_{im_i}1}}{\beta_{i0}\beta_{i1}^{x_{i1}}\beta_{i2}^{x_{i2}}\cdots\beta_{im_i}^{x_{im_i}1}+1} \\ &= \frac{\prod_{j=1}^{m_i}\beta_{i0}\beta_{ij}^{x_{ij}}}{\prod_{j=1}^{m_j}\beta_{i0}\beta_{ij}^{x_{ij}}+1}\,(i=1,\cdots 5)\,(m_i=20) \end{aligned}$$

(3) 约束条件：

本文同时还有隐含的约束条件，即根据变量的实际含义，找出其常规以及约定俗成的限制。比如，表示原子个数的变量应该是非负整数，而同一原子的最小原子类型 E-State 应该小于等于最大原子类型 E-State。我们将变量的限制条件所得集合记为 D，那么约束条件可以写为 $x \in D$。

(4) 双目标规划模型：

$$\max f_0$$
$$\max sign\,(f_1-1)+sign\,(f_2-1)+sign\,(1-f_3)+sign\,(f_4-1)+sign\,(1-f_5)$$
$$s.t.x\in D$$

7.2 基于预测模型的多目标优化模型的求解

由于目标一是最主要的目标，我们将双目标规划转化为但目标规划：

$$\max f_0$$
$$\max sign\,(f_1-1)+sign\,(f_2-1)+sign\,(1-f_3)+sign\,(f_4-1)+sign\,(1-f_5)\geq 3$$
$$s.t.x\in D$$

由于变量较多且实际意义较为复杂，无法准确获取自变量的取值范围，所得结果不甚满意，故未在论文中展示相关结果。

8 模型评价及展望

本文问题一利用三种特征筛选方式进行特征重要性评估，筛选出的特征更具有普适性。

本文问题二中首先用 XGBoost 以及 RF 进行预测，而后利用遗传算法对 XGBoost 和 RF 进行参数调优，构建 GA-XGBoost 和 GA-RF 模型进行预测，最后再用遗传算法确定 GA-XGBoost 和 GA-RF 的权重，将二者结合成为 GA-XG-RF 组合预测模型，模型的各项误差均达到最小，拟合优度达到最高。

本文问题三中通过训练构建的组合预测模型对 ADMET 性质进行预测，但由于分类问题中组合模型对数据较为敏感，且由于时间有限，未将其他模型加入组合模型之中。今后的研究可以考虑组合三个以上模型。

本文问题四中建立了优化模型，但由于变量较多且实际意义较为复杂，故未在论文中展示相关结果。

参考文献

[1] 韩建军，南少伟，郭呈周，李建平. 基于随机森林的粮仓气密性评价模型 [J]. 现代食品, 2018：187-190.

[2] 张春富, 王松, 吴亚东, 等. 基于 GA_Xgboost 模型的糖尿病风险预测 [J]. 计算机工程,2020,46(3):315-320

[3] 周恬恬. 基于百度指数和随机森林的上证综指预测 [D]. 重庆: 重庆大学,2020.
[4] 朱丽琴. 基于孤立森林的入侵检测方法研究 [D]. 黑龙江: 哈尔滨工程大学,2020.
[5] Kier, L. B., and Hall, L. H. (1976). Molecular connectivity in chemistry and drug research, (New York: Academic Press).
[6] Wildman, S. A., and Crippen, G. M. (1999). Prediction of Physicochemical Parameters by Atomic Contributions. [J] Chem Inf Comput Sci 39, 868-873.
[7] Ghose, A.K. and Crippen, G.M. , Atomic physicochemical parameters for three-dimensional structure-directed quantitative structure-activity relationships. I. Partition coefficients as a measure of hydrophobicity, Journal of Computational Chemistry, 1986, 7:565-577;
[8] Ghose, A.K. and Crippen, G.M. , Atomic physicochemical parameters for three-dimensional-structure-directed quantitative structure-activity relationships. 2. Modeling dispersive and hydrophobic interactions, Journal of Chemical Information and Computer Science, 1987, 27:21-35.

附录 A　Python 源程序

B.1　第 2 问程序

mip1.py

```
{
    "cells": [
    {
        "cell_type": "code",
        "execution_count": null,
        "id": "665d6313",
        "metadata": {},
        "outputs": [],
        "source": [
        "DNA_SIZE = 24\n",
        "POP_SIZE = 100\n",
        "CROSSOVER_RATE = 0.8\n",
        "MUTATION_RATE = 0.005\n",
        "N_GENERATIONS = 50\n",
        "W_XG_BOUND = [0, 1]\n",
        "W_RF_BOUND = [0, 1]\n",
        "test_y = np.array(val_y)\n",
        "\n",
        "\n",
        "def F(W_XG):\n",
        "    \n",
        "    y_t = test_y\n",
        "    xg = gaxg_y\n",
        "    rf = garf_y\n",
        "    W_XG = W_XG.reshape(100,1)\n",
        "    W_RF = (1-W_XG).reshape(100,1)\n",
        "    return ((W_XG * xg + W_RF * rf -y_t) ** 2).sum(
           axis =1).reshape(100,1)\n",
        "\n",
        "def get_fitness(pop): \n",
        "    W_XG,W_RF = translateDNA(pop)\n",
        "    pred = F(W_XG)\n",
        "    fitness = -(pred - np.max(pred)) + 1e-3\n",
```

```
"    return  fitness.reshape(100,)\n",
"\n",
"\n",
"def translateDNA(pop): # 表示种群矩阵，pop\n",
"    W_XG_pop = pop[:,1::2] # 奇数列表示X\n",
"    W_RF_pop = pop[:,::2] # 偶数列表示y\n",
"\n",
"    #pop:(POP_SIZE,DNA_SIZE)*(DNA_SIZE,1) --> (
   POP_SIZE,1)\n",
"    W_XG = W_XG_pop.dot(2**np.arange(DNA_SIZE)
   [::-1])/float(2**DNA_SIZE-1)*(W_XG_BOUND[1]-
   W_XG_BOUND[0])+W_XG_BOUND[0]\n",
"    W_RF = W_RF_pop.dot(2**np.arange(DNA_SIZE)
   [::-1])/float(2**DNA_SIZE-1)*(W_RF_BOUND[1]-
   W_RF_BOUND[0])+W_RF_BOUND[0]\n",
"    return W_XG,W_RF\n",
"\n",
"def crossover_and_mutation(pop, CROSSOVER_RATE =
   0.8):\n",
"    new_pop = []\n",
"    for father in pop:\n",
"        child = father\n",
"        if np.random.rand() < CROSSOVER_RATE:
            \n",
"            mother = pop[np.random.randint(POP_SIZE)
   ]    \n",
"            cross_points = np.random.randint(low=0,
   high=DNA_SIZE*2) 随机产生交叉的
   点 #\n",
"            child[cross_points:] = mother[
   cross_points:] \n",
"        mutation(child)\n",
"        new_pop.append(child)\n",
"\n",
"    return new_pop\n",
"\n",
"def mutation(child, MUTATION_RATE=0.03):\n",
```

```
"    if np.random.rand() < MUTATION_RATE:
                    \n",
"        mutate_point = np.random.randint(0, DNA_SIZE
   *2)  \n",
"        child[mutate_point] = child[mutate_point]^1
   #    \n",
"def select(pop, fitness):    \n",
"    idx = np.random.choice(np.arange(POP_SIZE), size
   =POP_SIZE, replace=True,\n",
"                           p=(fitness)/(fitness.sum
   ()) )\n",
"    return pop[idx]\n",
"\n",
"def print_info(pop):\n",
"    fitness = get_fitness(pop)\n",
"    max_fitness_index = np.argmax(fitness)\n",
"\n",
"    print(\"max_fitness:\", fitness[
   max_fitness_index])\n",
"    W_XG,W_RF = translateDNA(pop)\n",
"    print最优的基因型:
   (\"\", pop[max_fitness_index])\n",
"    print(\"(W_XG),(W_RF):\", (W_XG[
   max_fitness_index]),(1-W_XG[max_fitness_index]))\n
   ",
"    return ( W_XG[max_fitness_index],1-W_XG[
   max_fitness_index])\n",
"\n",
"\n",
"pop = np.random.randint(2, size=(POP_SIZE, DNA_SIZE
   *2)) #matrix (POP_SIZE, DNA_SIZE)\n",
"for _ in range(N_GENERATIONS)迭代:#代N\n",
"    W_XG,W_RF = translateDNA(pop)\n",
"    pop = np.array(crossover_and_mutation(pop,
   CROSSOVER_RATE))\n",
"    fitness = get_fitness(pop)\n",
"    pop = select(pop, fitness) 选择生成新的种群#\n",
```

```
        "print_info(pop)\n",
        "max_fitness_index = np.argmax(fitness)\n",
        "W_XG,W_RF = translateDNA(pop)\n",
        "xg_weight,rf_weight = W_XG[max_fitness_index],1-W_XG
           [max_fitness_index]\n",
        "gaxgrf_y = xg_weight * gaxg_y + rf_weight *  garf_y\
           n",
        "gaxgrf_y\n",
        "print(\"MSE:\", metrics.mean_squared_error(val_y,
           gaxgrf_y))\n",
        "print(\"R2:\", metrics.r2_score(val_y,gaxgrf_y))\n",
        "print(\"RMSE:\", np.sqrt(metrics.mean_squared_error(
           val_y,gaxgrf_y)))\n",
        "print(\"MAPE:\",mape(val_y,gaxgrf_y))\n",
        "print(\"MAE:\",metrics.mean_absolute_error(val_y,
           gaxgrf_y))"
        ]
    }
    ],
    "metadata": {
        "kernelspec": {
            "display_name": "Python 3",
            "language": "python",
            "name": "python3"
        },
        "language_info": {
            "codemirror_mode": {
                "name": "ipython",
                "version": 3
            },
            "file_extension": ".py",
            "mimetype": "text/x-python",
            "name": "python",
            "nbconvert_exporter": "python",
            "pygments_lexer": "ipython3",
            "version": "3.8.8"
```

```
        },
        "varInspector": {
            "cols": {
                "lenName": 16,
                "lenType": 16,
                "lenVar": 40
            },
            "kernels_config": {
                "python": {
                    "delete_cmd_postfix": "",
                    "delete_cmd_prefix": "del ",
                    "library": "var_list.py",
                    "varRefreshCmd": "print(var_dic_list())"
                },
                "r": {
                    "delete_cmd_postfix": ") ",
                    "delete_cmd_prefix": "rm(",
                    "library": "var_list.r",
                    "varRefreshCmd": "cat(var_dic_list()) "
                }
            },
            "types_to_exclude": [
            "module",
            "function",
            "builtin_function_or_method",
            "instance",
            "_Feature"
            ],
            "window_display": false
        }
    },
    "nbformat": 4,
    "nbformat_minor": 5
}
```