基于机器学习的药物选取和研发

摘要

为了治疗乳腺癌，研究人员需要挑选出对 $ER\alpha$ 有强抑制性、同时能在人体内安全代谢的化合物。在实验中实验人员会得到一系列作用于 $ER\alpha$ 靶标的化合物以及它们的分子描述符，而建立化合物的定量活性关系是挑选药物的关键。我们为了帮助研究人员设计合适的药物，做了以下工作。

首先，我们对给出的数据进行了分析和预处理。我们发现部分分子描述符的数值在给出的文件中一直为常数，此类数据对后续分析没有意义，因此我们按照 QSAR 标准流程推荐的办法将其剔除。此外，我们还发现表中部分药物的分子描述符的数值存在“离群点”，但经过验证这些离群点并不代表错误数值，因此不能对它们进行去除，为了降低离群值的影响，我们采用 Sigmoid 而非其他归一化方法处理数据。同时，我们还发现一些分子描述符间存在明显的相关关系，我们在后续工作中同样按照 QSAR 标准流程推荐的办法将其剔除，最终得到了我们的有效描述符集合 Ω_v。

然后，我们将药物活性作为叶节点、药物分子描述符视作划分标准构建了随机森林，接着采用基于不纯度的重要性排序——通过观察不同分子描述符加入随机误差后对结果准确度的影响，可以得到影响显著性的判准。同时在选中的指标间进行成对相关性分析，按照“后序剔除，顺位次取”的标准剔除完高相关性的指标后，我们最终选取了 minsssN、LipoaffinityIndex、MDEC-23 等二十个具有最显著影响的指标，我们将其命名为 Ω_o。

接着，在第二三问中，我们采用多层感知机 MLP 进行求解。利用拉特马赫复杂度和**定理 2**，我们证明了本题中采用 RELU 激活函数的 MLP 可以拟合分子描述符到活性和安全性指标的映射。接着我们分别设计了代表两个映射的神经网络，将各个药物的 Ω_o 的数值作为输入、活性/安全性指标作为输出，并将已知数据划分为训练集和验证集进行训练，最终我们在验证集的的精度达到了 92.2%，我们也用此网络预测了 test 集中 50 个药物的相关性质。

最后，我们利用之前求得的神经网络模型，采用差分进化优化算法，寻找在整个 Ω_o 空间中符合安全标准的最大活性药物，并得出了各个分子描述符的取值范围，如我们求得的 C1SP2 可能取值就在 0.00-19.98 之间。

综上，我们在挑选出了有着最显著影响的描述符集合 Ω_o 后，采用神经网络预测了 Ω_o 和药物活性/安全性指标的关系，并寻找到了符合入药标准的最大活性药物具有的 Ω_o 的范围，对乳腺癌药物的研发做出了贡献。

关键字：随机森林　多层感知机　差分进化算法

目录

一、 问题重述

为了治疗乳腺癌，研究人员须挑选能抑制 $ER\alpha$ 且具有良好安全性（用 ADMET 描述，本题中有五个指标）的药物。现在给出了 1974 个潜在药物和它们的 729 个描述符，并已知它们的对 $ER\alpha$ 的生物活性值和五个安全性指标。

1.1 问题的提出

问题一：挑选出前二十个对 $ER\alpha$ 的生物活性值有显著影响的分子描述符。

问题二：利用以上二十个分子描述符，构建化合物对 $ER\alpha$ 的生物活性的定量预测模型，并对 50 个给出化合物的生物活性进行预测。

问题三：利用 729 个分子描述符构建五个安全性指标的分类模型，并预测给出的 50 个化合物的安全性。

问题四：寻找合适的分子描述符，给出它们合适值的范围，使得此时化合物对 $ER\alpha$ 的生物活性高，同时具有良好的安全性。

二、 模型的假设

- 表格中所给药物足够代表整个药物集体，具有良好的普遍意义。
- 药物活性和安全性与且仅与表格中的 729 个分子描述符有关。
- 训练集中的药物的性质可以很大程度上反映测试集中的药物。
- 表中所给数据均是合理无误的。(后续我们会进行论证说明这个假设的合理性和必要性)

三、 符号说明

符号	意义
Ω	分子描述符
D	潜在药物
pIC_{50}	生物活性值
I_j	第 j 个指标的重要性
$r(a)$	神经网络损失期望
$\mathcal{N}$	神经网络函数集合
$\mathcal{W}$	神经网络宽度
$\mathcal{D}$	神经网络深度
$\mathcal{L}$	神经网络损失函数

四、 数据预处理

4.1 常值分析

在 QSAR 建模的数据预处理过程中，常值（Constant Value）去除是常用的数据清洗方法之一。这是由于某些分子描述符值在任意一个潜在药物中都相同，这些分子描述符对于后续分析没有任何意义，因此为了简化分析，我们通常会将其去掉。换言之，我们需要找到一个集合 Ω_N，其有：

$$\begin{aligned} &\Omega_N \in \Omega_{total}, \\ &s.t. \forall\, \Omega_i \in \Omega_N,\, \Omega \equiv C_i, \end{aligned} \tag{1}$$

我们将这个集合 Ω_N 从分子描述符集合 Ω_{total} 中去掉，即完成了常值（Constant Value）去除。在本题中，我们发现对于 1974 个潜在药物，nB、nBondsQ 等分子描述符的值恒为常值，因此我们将此类分子描述符去掉不纳入考虑，最终剩余 504 个分子描述符，我们将其命名为 Ω_v。

4.2 数据归一化

我们观察到虽然表中有极端数据出现，但这并不代表数据本身有误，如分子描述符 nP（磷原子个数），1 到 1974 药物中该分子描述符的值如下图所示：

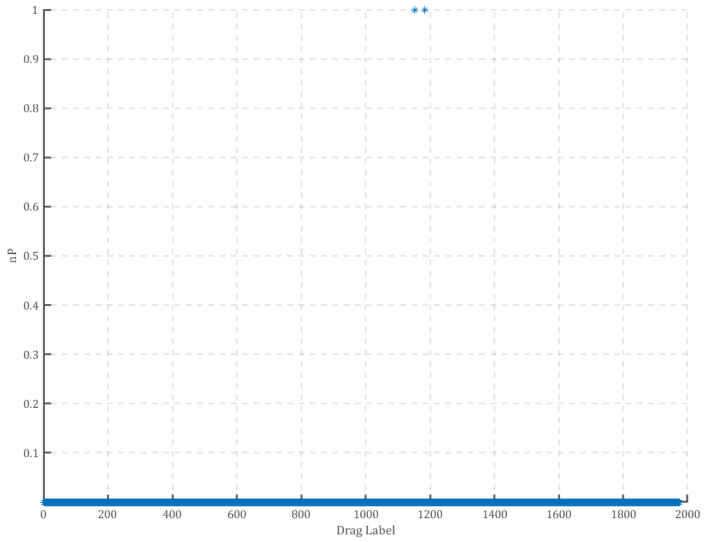

图 1　分子描述符协方差矩阵（局部）

能明显观察到有两三个“异常值”，但我们检查这些“异常值”后发现发现：尽管绝大部分药物没有磷原子，确实有极少数药物含 1 到 2 个磷原子，即根本不存在所谓的异常值。

换言之，我们无法通过寻找离群值的方法对数据进行清洗（因为某些药物因其自身特性而具有特殊的分子描述特征值），但出现的“异常值”却会“抑制”其他正常数值的表达，进而对我们后续神经网络等算法产生影响，为了在保留特征药物的同时，降低其对其他药物的相应值的影响，我们采用 sigmoid 函数对分子描述符进行归一化。

$$\Omega_1 = \frac{1}{1 + e^{-\Omega_0}}, \tag{2}$$

4.3 成对相关性分析

QSAR 另一常用的数据预处理方法即是成对相关性分析（Pair-wise Correlation），在本题中，由于许多指标均为属于同一大类描述符下的细节描述符，因此其之间会有极强的关联性，而这类关联性会产生大量的冗余，而且在第一问选择 20 个关键描述符时，我们也不想多次选到同一大类的描述符，因此我们首先计算了上述 504 个分子描述符的协方差矩阵，如**图 2**所示（仅展示部分）。

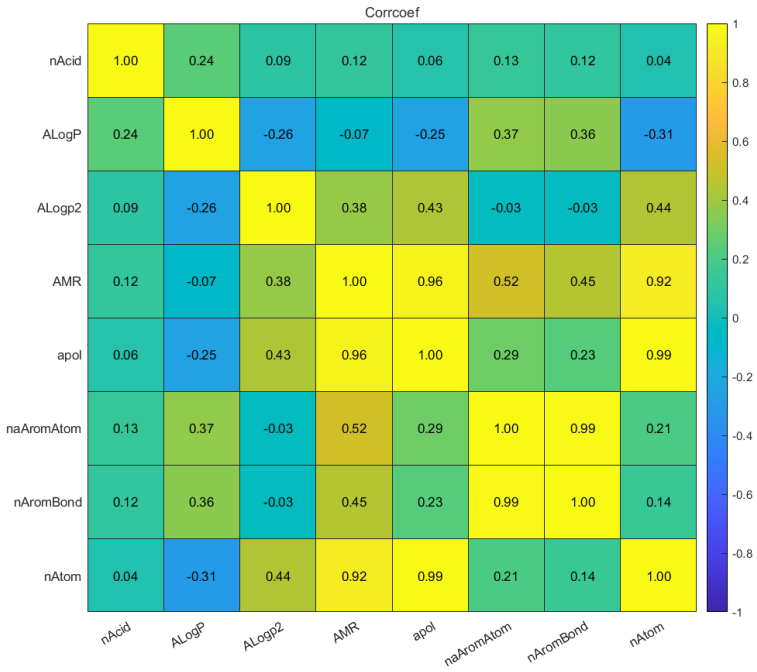

图 2　分子描述符协方差矩阵（局部）

图中第一到八号分子描述符分别为 nAcid、ALogP、ALogp2、AMR、apol、naAromAtom、nAromBond、nAtom 的指标值。可以看到 naAromAtom（芳香原子数）和 nAromBond（芳香键数）相关性高达 0.99，事实上二者均是衡量药物芳香性的指标，理应拥有高相关性。但由于数据特性，有些分子描述符虽然在化学层面上不具有强相关性，但在给出的 1974 个潜在药物分子描述符指标却有强相关性。比如上图中的 AMR 指标（摩尔分裂性）和 apol（潜在极性分子数），因此直接对原数据进行成对相关性分析并剔除冗余是不可行的，而逐个判断两个数据高相关性指标的化学相关性又过于耗时，所以我们会在决定了 20 个高显著性的分子描述符后再进行成对相关性分析，详见下一节。

五、 选取分子描述符

由于各个分子描述符和对 $ER\alpha$ 的生物活性不一定为线性相关，且与生物活性相关性强的描述符不一定是能产生显著影响的描述符，直接计算各描述符与生物活性指标的相关系数再通过排序选择前二十个的思路未免有些欠妥。

顺着显著特征选择类型的思路，我们采用随机森林算法对此题进行建模。

5.1 模型的建立

我们的随机森林由若干回归树组成，每一棵回归树都表述了一个回归结构，换言之，即用树模型做回归问题，每一片叶子都代表某种情况下响应变量的一个预测值，最

终预测结果将由所有回归树给出的值取平均数得到。

树的构建包括两个部分：样本和特征。在本题中，我们将分子描述符 $\Omega_1, \ldots, \Omega_v$ 视作训练集的特征，总训练集由 1974 个药物 $D_1, \ldots, D_{1974}$ 组成，每次构建回归树我们会有放回地抽取 m 个训练集中的样本进行训练，每棵回归树的最终输出为生物活性值 pIC_{50} 。

我们按照下列步骤构建每一棵回归树（以最小二乘回归树生成算法为例）：

- 输入训练数据集 $D_1, \ldots, D_m$，在训练集所在输入空间中，递归地将每个区域划分为两个子区域并决定每个子区域上的输出值。
- 划分方法为：选择最优的切分变量 j（即样本的特征，本题中为分子描述符）和切分点 s（分割处的取值），使得：

$$\min_{j,s}\left[\min_{c_1}\sum_{x_i\in R_1(j,s)}(y_i-c_1)^2+\min_{c2}\sum_{x_i\in R_2(j,s)}(y_i-c_2)^2\right], \tag{3}$$

  。

  式中，x_i, y_i 分别代表第 i 个样本特征向量和样本的值，$R_1(j,s)$ 代表由 j 和 s 切分出的区域，c_m 表示第 m 片叶子的预测值，为了最小化总体的 MSE，只需要最小化每一片叶子的 MSE, 而最小化一片叶子的 MSE，只需要将预测值设定为叶子中含有的训练集元素的均值。因此我们将 c_m 设定为：

$$c_m=\frac{\sum y_i|x_i\in R_1(j,s)}{k}, \tag{4}$$

  式中，k 为落在 $R_1(j,s)$ 中样本个数。

  在每次划分的时候，我们致力于最小化各个叶子节点的 MSE 之和，这里采用启发式的方法，遍历所有的切分变量和切分点，然后选出叶子节点 MSE 之和最小的那种情况作为划分。因此遍历变量 j，对固定的切分变量 j 扫描切分点 s，选取能让**式 3**达到最小的 (j,s)。
- 找到最优的切分点 (j,s) 后，我们即可将输入空间划分为两个区域，接着对每个区域重复上述划分过程，直到满足停止条件为止。然后我们将每个区域内样本值的众数作为各区域对应叶节点的输出值。这样即得到了一颗能将整个输入空间划分为 $R_1, R_2, \ldots, R_m$ 共 m 个区域的回归树。
- 重复上述三个步骤，不断抽取样本构建回归树，生成随机森林。

而生成随机森林后，我们采用基于不纯度的重要性排序，对每一颗决策树，选择相应的没有参与决策树训练的袋外数据（OOB）计算其生物活性值，并计算出其与标准值的误差，记为 err_1。接着我们随机对袋外数据 OOB 所有样本的特征 j 加入高斯噪声干扰（可以随机改变样本在特征处的值），再次计算袋外数据误差，记为 err_2 。假设森林中有 N 棵树，则我们引入变量 I_j:

$$I_j = \frac{\sum(err_2 - err_1)}{N}, \tag{5}$$

若加入随机噪声后，袋外数据准确率大幅度下降，即加噪声后预测值的误差 err_1 与加噪声之前预测值的误差 err_2 之差 I_j 变大，则说明特征 j 对于样本的预测结果有很大影响，进而说明重要程度较高。故我们可用 I_j 衡量特征 j 的重要性。

得到各个特征的重要性指标 I_j 后，我们选取排名前二十的指标作为第一题的解。

5.2 模型的求解

经过我们对随机森林的求解，我们得到了排名靠前的分子描述符，如**图 3**所示（仅列出前三十名）：

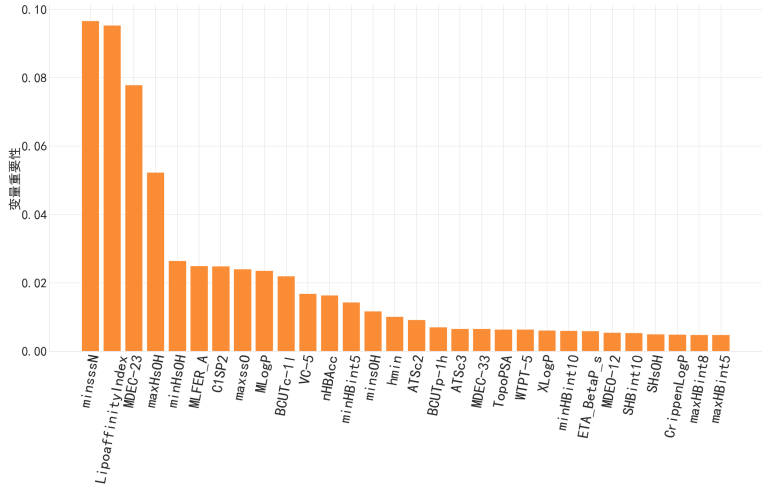

图 3　最显著的分子描述符

由图知，对生物活性影响最显著的分子特征前二十名分别为 minsssN、LipoaffinityIndex、MDEC-23、maxHsOH、minHsOH、MLFER_A、C1SP2、maxssO、MLogP、BCUTc-1l、VC-5、nHBAcc、minHBint5、minsOH、hmin、ATSc2、BCUTp-1h、ATSc3、MDEC-33、TopoPSA（按先后次序排名）。

我们计算其相关系数矩阵，并将其用热力图表示如**图 4**所示：

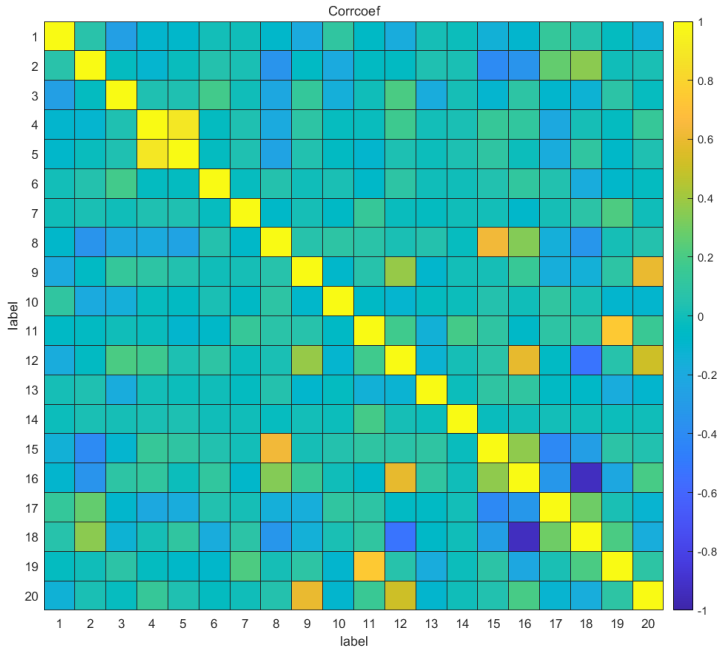

图 4　显著分子描述符相关系数矩阵

接着我们查找其中相关系数的绝对值大于 0.8 的分子描述符，发现有两组分子特征不满足要求，在上图中我们也能看到，第五排第四列，代表 maxHsOH 和 minHsOH（最大羟基氢个数和最小羟基氢个数）的相关系数，其值高达 0.896，呈明显正相关；第十六排第十八列，代表 ATSc2 和 ATSc3 分子描述指标的相关系数，其值为-0.936，呈明显负相关。

因此我们在第二问选最显著的 20 个分子描述符时，将去掉两组中排序相对靠后描述符：minHsOH 和 ATSc3，并顺次取第二十一位和第二十二位分子描述符 WTPT-5 和 XLogP，经过验证，修改完后的相关系数矩阵每一项都小于 0.8，满足要求，因此我们后续将使用 minsssN、LipoaffinityIndex、MDEC-23、maxHsOH、MLFER_A、C1SP2、maxssO、MLogP、BCUTc-1l、VC-5、nHBAcc、minHBint5、minsOH、hmin、ATSc2、BCUTp-1h、MDEC-33、TopoPSA、WTPT-5、XLogP 作为选取的分子描述符，我们将其命名为 Ω_o。更新后的相关系数矩阵如**图 5**所示。

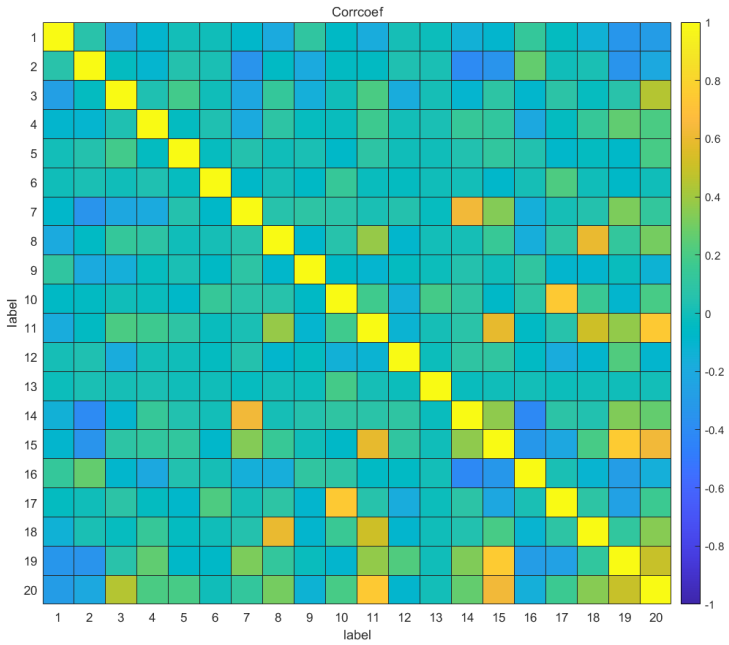

图 5　更新后的显著分子描述符相关系数矩阵

六、 预测药物的 $ER\alpha$ 生物活性

6.1 模型的建立

为了建立前二十个显著的分子描述符 Ω_o 和 $ER\alpha$ 生物活性的关系，我们采用机器学习的方法。我们将训练集设为

$$S = \{z_1 = (x_1, y_1), z_2 = (x_2, y_2) \ldots, z_n = (x_n, y_n)\}, \tag{6}$$

其中自变量设置为分子描述符的值 $x_i \in \mathbb{R}^{20}$，因变量设置为药物生物活性的值 $y_i \in \mathbb{R}$，$i = 1, 2, \ldots 1974$

假定我们的化合物活性 y_i 与分子描述符 x_i 满足某一函数关系 $y = f(x), f \in \mathcal{B}$。现在确定一个损失函数 $\mathcal{L} : \mathcal{B} \times z \longrightarrow \mathbb{R}_+$，令 $r : \mathcal{B} \longrightarrow \mathbb{R}_+$

即：

$$r(a) = \mathbb{E}[\mathcal{L}(a, z)], \tag{7}$$

那么我们要求的即是：

$$f = \arg\min_{a \in \mathcal{B}} r(a), \tag{8}$$

现在我们选取神经网络函数逼近真实解 $f(x)$。设 $\mathcal{N}$ 是神经网络函数的集合，$\mathcal{N} \subseteq \mathcal{B}$

那么我们的目标即转化为了：

$$f = \arg\min_{a \in \mathcal{N} \subseteq \mathcal{B}} r(a), \tag{9}$$

我们的神经网络选取为 $\mathbb{R}^{20} \longrightarrow \mathbb{R}$

$$\begin{aligned} &f_0(x) = x, \\ &f_l(x) = Q_l(A_l f_{l-1} + b_l), \\ &\vdots \\ &f_{\mathcal{N}} = f_{\mathcal{D}}(x) = A_{\mathcal{D}} f_{\mathcal{D}-1} + b_{\mathcal{D}}, \end{aligned} \tag{10}$$

其中 $A_l \in \mathbb{R}^{N_l \times N_l - 1}$ 和 $b_l \in \mathbb{R}^{N_l}$ 代表神经网络第 l 层的参数，Q_l 为激活函数，我们选用 RELU 激活函数。

$\mathcal{D}$ 为网络的深度，网络宽度 $\mathcal{W}$ 可以用下式表示：

$$\mathcal{W} = \max\left\{N_1, N_2, \ldots N_{\mathcal{D}}\right\}, \tag{11}$$

$\sum_{l=1}^{\mathcal{D}} N_l$ 为网络中的总单元数，$\phi = \{A_l, b_l\}$ 是每一层的参数。

值得注意的是，我们此时并不知道训练集在整个空间的分布，也即 z 的概率分布，所以无法用 $\mathbb{E}[\mathcal{L}(a, z)]$ 计算得到 $r(a)$ 因此我们用经验损失替代，如下式：

$$\hat{r}(a) = \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(a, z_i), \tag{12}$$

最终我们求解的模型即为：

$$\widehat{f}_{\mathcal{N}} = \arg\min_{a \in \mathcal{N} \subseteq \mathcal{B}} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(a, z), \tag{13}$$

此处我们采用 SGD 梯度随机下降法进行优化，会得到一个随机解 (Random Solver)$S \in \mathcal{N}$

我们最后对上述算法进行误差估计，原误差 $r(S) - r(f)$ 可以按照下式可以分解成统计误差 statistical error（在神经网络构建中我们仅使用了经验损失）、优化误差 optimization error（在随机梯度下降算法中由于迭代次数有限，我们并未得到真实最优解）和函数逼近误差 approximation error（神经网络无法完全拟合原函数），并将其表示如下：

$$\begin{aligned} r(S) - r(f) &= r(S) - \widehat{r}(S) + \widehat{r}(S) - \widehat{r}(\widetilde{f_{\mathcal{N}}}) + \widehat{r}(\widetilde{f_{\mathcal{N}}}) - \widehat{r}(f_{\mathcal{N}}) + \widehat{r}(f_{\mathcal{N}}) - r(f_{\mathcal{N}}) + r(f_{\mathcal{N}}) - r(f), \\ &\leq \underbrace{r(f_{\mathcal{N}}) - r(f)}_{approximation\ \ error} + \underbrace{2 \sup_{a \in \mathcal{N}} |r(a) - \widehat{r}(a)|}_{statistical\ \ error} + \underbrace{\widehat{r}(S) - \widehat{r}(\widetilde{f_{\mathcal{N}}})}_{optimazation\ \ error}, \end{aligned} \tag{14}$$

对上式两边分别取期望，得到下述估计：

$$\mathbb{E}r(S) - r(f) \leq r(f_{\mathcal{N}}) - r(f) + 2\mathbb{E}\sup_{a\in\mathcal{N}}|r(a) - \widehat{r}(a)| + \mathbb{E}\widehat{r}(S) - \widehat{r}(\widetilde{f_{\mathcal{N}}}), \tag{15}$$

在本文中，我们忽略优化误差，即认为 $\mathcal{E}_{app} = 0$，下面我们分别估计函数逼近误差和统计误差的上界。下面我们将证明使用机器学习拟合回归函数是可行的。

- 统计误差上界估计:

定理 1 利用拉特马赫复杂度，*VC dimension* 给出统计误差的上界：

$$2\sup_{a\in\mathcal{N}}\mathbb{E}\,|r(a) - \widehat{r}(a)| \leq 4\sqrt{\frac{2VC(\mathcal{N})log(en/VC(\mathcal{N}))}{n}}, \tag{16}$$

其中 n 为训练样本数量，$VC(\mathcal{N})$ 为 $ReLU$ 网络的 *VC dimension*，其大小随网络宽度 W 和网络深度 D 变化而变化。

proof:

$$\mathbb{E}\sup_{a\in\mathcal{N}}|r(a) - \widehat{r}(a)| \leq 2\mathbb{E}[\mathcal{R}ad(\mathcal{L}\circ\{z_1, z_2, ..., z_n\})], \tag{17}$$

其中：

$$\mathcal{L} = \{z \in \mathbf{Z} \to l(a, z) \in \mathbb{R} : a \in \mathcal{N}\}, \tag{18}$$

对于拉特马赫复杂度，我们有如下不等式：

$$\mathcal{R}ad(\mathcal{A}\circ x) \leq \sqrt{\frac{2VC(\mathcal{A})log(en/VC(\mathcal{A}))}{n}}, \tag{19}$$

结合上述两式即可得到定理结果：

$$2\sup_{a\in\mathcal{N}}\mathbb{E}\,|r(a) - \widehat{r}(a)| \leq 4\sqrt{\frac{2VC(\mathcal{N})log(en/VC(\mathcal{N}))}{n}}, \tag{20}$$

- 函数逼近误差：

定理 2 *若函数 $f \in C^s([0,M]^d)$，那么 ReLU 神经网络对其有如下逼近性质：*

$$r(f_{\mathcal{N}}) - r(f) = \mathcal{O}\left(\|f\|_{C^s([0,M]^d)} \mathcal{D}^{-2s/d} \mathcal{W}^{-2s/d}\right), \tag{21}$$

其中 $\mathcal{D}$，$\mathcal{W}$ 是我们采用的神经网络的深度和宽度，d 是样本点的维数，由于此处我们选择了 20 个特征指标，此处 $d = 20$。

最后我们给出随机解与真实解之间的误差：

$$\begin{aligned} &\mathbb{E}r(S) - r(f), \\ &\leq \varepsilon_{sta} + \varepsilon_{app} + \varepsilon_{opt}, \\ &\leq 4\sqrt{\frac{2VC(\mathcal{N})log(en/VC(\mathcal{N}))}{n}} + \mathcal{O}\left(\|f\|_{C^s([0,M]^d)} \mathcal{D}^{-2s/d} \mathcal{W}^{-2s/d}\right), \end{aligned} \tag{22}$$

该估计说明了——通过调整网络宽度 W 和网络深度 D，我们可以使机器学习求解得到的人工神经网络与真实解之间差异在一个可控范围之内。这同时也证明了我们的模型是可行的。

6.2 模型的求解

我们运用 python 的 pytorch 包进行神经网络的搭建，考虑到本题作为一个回归类型的题目，我们设计了 MLP 全连接神经网络进行训练。我们的网络由输入层（宽度为 20）、隐藏层（4 层）和输出层（宽度为 1）组成，选用 RELU 激活函数，通过输入各个药物的分子描述符，然后计算网络输出的活性值和真实活性值的差异，将该值设置为损失函数 $\mathcal{L}$ 并反向传播，采用 SGD 优化方法，最终优化得到我们的网络模型。

训练过程中验证集的 loss 变化如**图 6**所示：

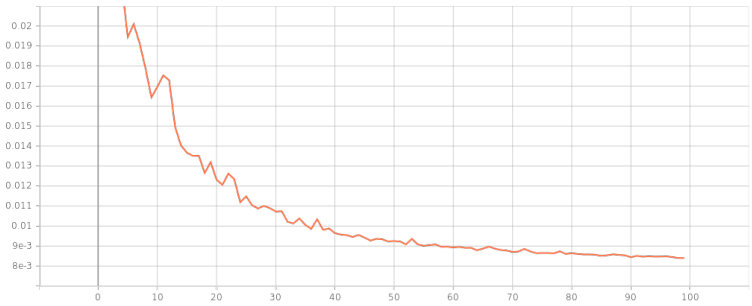

图 6　loss 随训练推进的变化情况

最后我们的回归结果如**图 7**，可见我们的神经网络成功预测了活性变化的趋势。

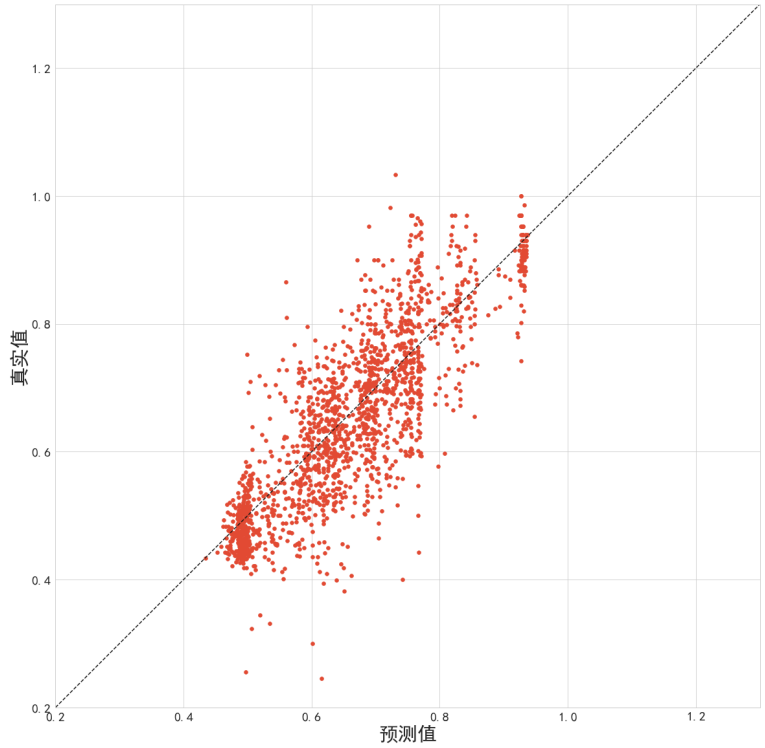

图 7　MLP 回归结果

同时我们也预测出了 test 集中 50 个药物的活性，结果为：0.663、0.726、0.735、0.728、0.702、0.633、0.644、0.637、0.728、0.645。（仅列出前十个）

七、 预测药物的安全性

7.1 模型的建立和求解

与第二问类似，我们仍采用多层感知机 MLP 对该问题进行求解，不同的是，我们在网络输出层输出时用 sigmoid 函数进行归一化。意在将网络的输出设置为各个安全指标合格的概率，用 $P_s(P_s \in [0,1])$ 表示，然后我们将损失函数 $\mathcal{L}$ 设置为某安全指标合格概率和该药物安全与否的真实情况（概率为 1 或 0）作差，以次训练我们的网络。

最终我们将 test 集的 50 个药物放入模型中预测，若预测出来安全的概率（输出）大于 50%，我们则认为药物是安全的。以微核试验通过与否为例，50 个药物得到的结果为：1、1、1、1、1、0、0、0、1、1、1、1、1、1、1、1、1、1、1、1、1、0、0、1、1、0、1、1、1、1、1、1、1、1、1、1 、0、1、1、1、1、1、1、1、1、1、1、1、0。其中 1 代表有遗传毒性，0 代表没有遗传毒性。

7.2 模型的验证

仅用单一方法得到的分类结果不一定可靠。为了最大可能地增强预测准确性，我们还采用了支持向量机和随机森林对药物安全性进行分类。最终支持向量机的分类精度在验证集上最高达到 91.9%，SVM 训练过程和结果如**图 8**所示。

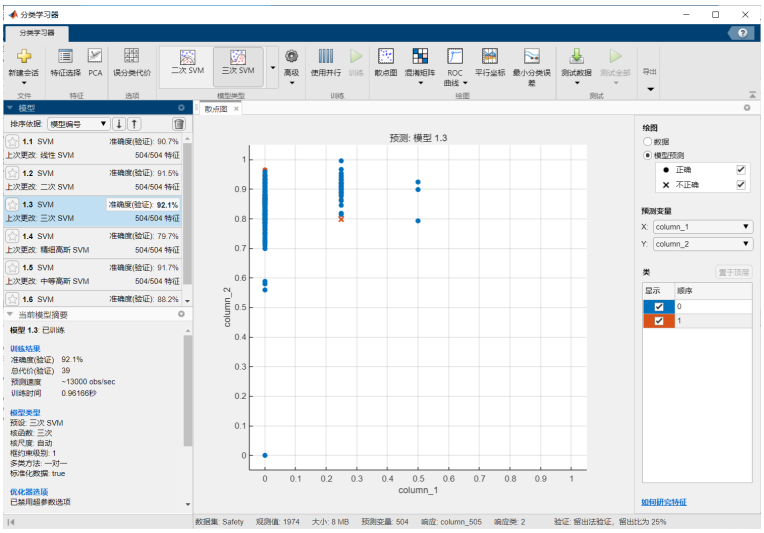

图 8　svm 训练过程

最终测试集得到的混淆矩阵如**图 9**所示：

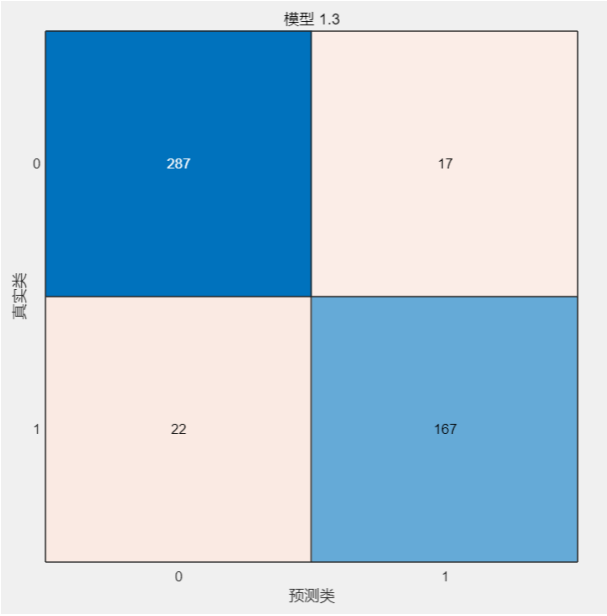

图 9　混淆矩阵

MLP 训练过程中训练集的二分类交叉熵损失变化如**图 10**所示。

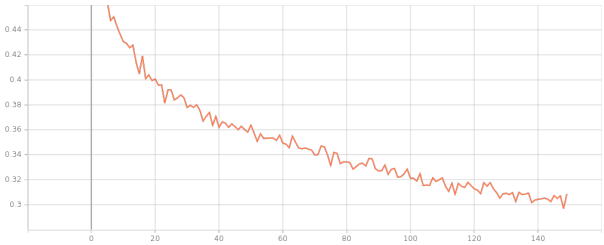

图 10　分类训练过程损失函数的变化

最终随机森林得到的分类精度为 Caco-2 安全指标:85.68%、CYP3A4 安全指标:92.22%、hERG 安全指标:86.98%、HOB 安全指标:82.02%、MN 安全指标:89.01%。

我们最终药物安全性的结果采用三个算法投票的方式，根据少数服从多数原则确定药物是否具有某一安全性特征，以微核试验通过与否为例，50 个药物中后二十个得到的结果为：1、1、1、1、1、1、0、1、1、1、1、1、1、1、1、1、1、1、0。其中 1 代表有遗传毒性，0 代表没有遗传毒性。

八、 寻找最优分子指标

8.1 模型的建立

在第二问得到的网络的基础上，我们采用遗传算法的变种——差分进化算法（后续我们简称为 DE 算法）对该问题进行优化，算法流程如下：

- 初始化：随机初始化 N_p 个 D 维参数向量 x，$x(i)$ 表示第 i 个解，每个解参数可以表示为 $x(i,j)$，其中 $i = 1, 2, \ldots, N_p, j = 1, 2, \ldots, D$
  解数目 N_p 根据情况选择，本文选取 $N_p \in [50, 200]$。
- 变异：变异的目的是防止进化过程陷入局部最优解，而无法得到全局最优解。对于每个解向量 $x(i)$，对应的变异向量 v 可以表示为：

$$v(i) = x(r_0) + F * (x(r_1) - x(r_2)), \tag{23}$$

  其中 $r0, r1, r2$ 为属于 $[1, \ldots, N_p]$ 的三个随机数，并且 i, r_0, r_1, r_2 都不相同，这要求 N_p 必须大于等于 4。变异算子 F 取值范围为 $[0, 2]$，F 过小可能陷入局部最优，F 过大则不容易收敛。
  变异以后的值 $v(i,j)$ 若超出了边界，我们会将 $v(i,j)$ 设置为边界值。
- 交叉：接下来求交叉向量 u，对于每个 u 的每个维度上的值，有：

$$\begin{aligned} u(i,j) &= v(i,j), if\ rand() \le CR, \\ u(i,j) &= x(i,j), if\ rand() > CR, \end{aligned} \tag{24}$$

$rand()$ 是一个随机数，CR 是交叉算子，且有 $CR \in [0,1]$，它被用来控制选择变异向量值还是原来的向量值。

- 选择：把交叉向量和原向量的函数值作对比，也即对比 $f(u(i))$ 和 $f(x(i))$ 哪个更优，选择较优者，更新向量 x，进行下一步。
- 终结条件：当最后的解满足条件，或者遍历次数达到最大，则结束，否则重复步骤 2 到 4 步骤。

8.2 模型的求解

在 DE 算法实现的过程中，我们将之前求得的 20 个分子描述符作为输入的参数向量 x，随机变量数 N_p 设置为 2000，变异系数设置为 0.8。

在选择步骤中，挑选更优的向量时，我们将第二问中神经网络输出的相反数设置为函数值，而且除了优先挑选出活性更高的分子描述符向量，考虑到选择的药品还应当具有良好的 ADMET 性质（至少三个指标安全），我们在进行选择的时候仅挑选满足至少三个良好 ADMET 性质的变异项。在经过 1000 轮迭代之后，所有向量均会逐渐收敛到最优解附近，我们挑选出所有满足 ADMET 要求的向量，并计算出它们的范围，以作为我们所求的分子描述符最优区间。最终，我们得到的范围如下表所示：

分子描述符	下界	上界
minsssN	0.0004	2.7340
LipoaffinityIndex	-4.5860	22.9921
MDEC-23	0.0319	54.0063
maxHsOH	0.0001	0.8521
MLFER_A	-0.8518	7.7528
C1SP2	8.3603e-05	19.9813
maxssO	0.0007	6.7248
MLogP	1.4628	5.7498
BCUTc-1l	-0.4217	-0.1888
VC-5	0.00253	1.4801
nHBAcc	0.0360	65.9984
minHBint5	-1.1907	11.5502
minsOH	0.0135	11.7261
hmin	-0.5883	0.3871
ATSc2	-2.3764	-0.0157
BCUTp-1h	7.9733	16.7410
MDEC-33	0.0060	49.7532
WTPT-5	0.0122	125.5552
TopoPSA	12.9659	1206.2913
XLogP	-3.5819	14.2780

九、 模型的优缺点

9.1 模型的优点

1. 本文根据数据特征，对数据做了针对性的清理工作，提高了后续处理的效率和精度。
2. 本文认真考虑了显著性的意义，找出了对活性最显著的分子指标。

3. 本文采用神经网络算法，较好地模拟了安全性和活性的映射函数。
4. 本文运用 DE 优化算法，让随机分布的向量收敛到顶点附近，找到了全局最优解。

9.2 模型的缺点

1. 由于数据量较小，本文神经网络的泛化性无法得到保障。
2. 本文仅用单一方法求取结果，准确性欠佳。
3. 在利用随机森林计算重要性时，本文没有采用最优的 GINI 算法。

参考文献

[1] Song S, He R, Shi Z, et al. Variable Importance Measure System Based on Advanced Random Forest[J]. Computer Modeling in Engineering & Sciences, 2021, 128(1): 65-85.