# 统计分析与建模大作业

小组成员：

1854062 许之博

1. **问题1**

对数据进行预处理并提取与pIC50相关性较强的特征。

1. **数据预处理：**
2. **import** pandas as pd
3. file1\_path = 'f1.csv'  #化合物特征及实验数据
4. f1 = pd.read\_csv(file1\_path)
   1. **缺失值：**
5. **print**(f1.columns[f1.isnull().any()].tolist())  # []

经验证，数据中不存在缺失值。

* 1. **异常值：**

1. **from** scipy **import** stats
2. **from** collections **import** Counter
3. fz =pd.DataFrame()
4. **for** attr **in** f1.columns[1:]:
5. fz[attr+'\_zscore']=stats.zscore(f1[attr])
6. list=[]
7. **for** index, row **in** fz.iterrows():
8. **for** attr **in** fz.columns:
9. **if** abs(row[attr])>=3:
10. list.append(index)
11. result=Counter(list)
12. list2=result.most\_common(52)
13. **for** i **in** list2:
14. f1.drop(index=[i[0]],inplace=True)
15. fz.drop(index=[i[0]],inplace=True)

此处使用zscore来判断存在异常值的化合物，并将其从数据中剔除。zscore，也就是标准分数，它是将原始分数与团体的平均数之差除以标准差所得的商数，是以标准差为单位度量原始分数离开其平均数的分数之上多少个标准差，或是在平均数之下多少个标准差。

首先，计算每条数据各个属性值的zscore值。当某个属性值的zscore值大于等于3时，认为该属性值异常。统计出含有异常属性值的数据，并统计含有异常属性值的个数，对筛选出的异常数据按含有异常属性值的数量降序排序。最后，将含有超过30个异常属性的数据删除。

**1.3 重复值：**

对于可能存在的重复值情况，存在行重复与列重复两种情况。

**1.3.1 行重复**

SMILES是使用ASCII字符明确描述分子结构的规范，SMILES使用一串字符来描述一个三维化学结构。典范SMILES保证每个化学分子只有一个SMILES表达式，典范SMILES常用于分子数据库的索引。因此，可以使用SMILES属性来判断是否存在行重复情况，如存在两行数据的SMILES相同的情况，那么可以判断为行重复。

1. f1 = pd.read\_csv(file1\_path)
2. print(f1.shape) # (1974,736)
3. f1.drop\_duplicates(subset='SMILES')  # 判断是否有行重复
4. print(f1.shape) # (1974,736)

经验证，不存在行重复的情况。

* + 1. **列重复**

列重复判断的目的是判断是否存在属性重复的情况。

1. columns = f1.columns.values  # 判断是否有列重复
2. print(len(columns)) # 736
3. np.unique(columns)
4. print(len(columns)) # 736

经验证，不存在列重复的情况。

1. **相关性分析：**

使用预处理后得到的数据，提取与pIC50相关性较强的属性。IC50指半抑制

浓度(或称半抑制率),对指定的生物过程抑制一般时所需的药物或抑制剂浓度。pIC50=-log(IC50)。

当两个连续变量呈现出线性相关或满足正态性，一般使用Pearson相关系数，当数据为定量数据，且不服从正态性则使用Spearman相关系数。此处使用Spearman相关系数进行相关性分析。

1. f1\_corr = f1.corr('spearman')['pIC50'].dropna()

.sort\_values()

1. f1\_corr\_sort = f1\_corr.abs().sort\_values()

|  |  |  |
| --- | --- | --- |
| 编号 | 特征名 | 相关系数 |
| 1 | MDEC-23 | 0.552716 |
| 2 | MLogP | 0.548946 |
| 3 | LipoaffinityIndex | 0.530547 |
| 4 | C1SP2 | -0.511839 |
| 5 | maxsOH | 0.495676 |
| 6 | nC | 0.485972 |
| 7 | CrippenLogP | 0.461918 |
| 8 | VP-5 | 0.466664 |
| 9 | ATSp4 | 0.461388 |
| 10 | minsOH | 0.458949 |

表1. 通过相关性分析筛选出的10个特征

在所有729个特征中，与pIC50相关性较强的是以下20个：

1. ['ATSp3', 'C2SP2', 'VP-3', 'SwHBa', 'ATSP-1',
2. 'ATSp2', 'VP-4', 'BCUTp-1h', 'AMR', 'ATSp5', 'minsOH',
3. 'ATSp4', 'VP-5', 'CrippenLogP', 'nC', 'maxsOH',
4. 'C1SP2', 'LipoaffinityIndex', 'MLogP', 'MDEC-23']
5. **问题2**

**1. 训练集和测试集划分**

使用问题一种筛选出来的20个与pIC50相关性较强的特征，按照4：1划分训练集和测试集。

1. attributes = corr.get\_attributes()  # 在问题一中筛选出的特征
2. f1\_attr20 = f1[attributes]
3. f1\_attr20 = shuffle(f1\_attr20)
4. X\_train,X\_test,Y\_train,Y\_test = train\_test\_split(f1\_attr20[attributes[:20]],f1\_attr20['pIC50'],train\_size=0.80)

**2. 构建pIC50的回归模型**

利用训练集训练一个多元线性回归模型。此处选用最小二乘法构建回归模型。

1. **import** statsmodels.api as sa
2. model2 = sa.OLS(Y\_train,X\_train)

**3. 模型概览**

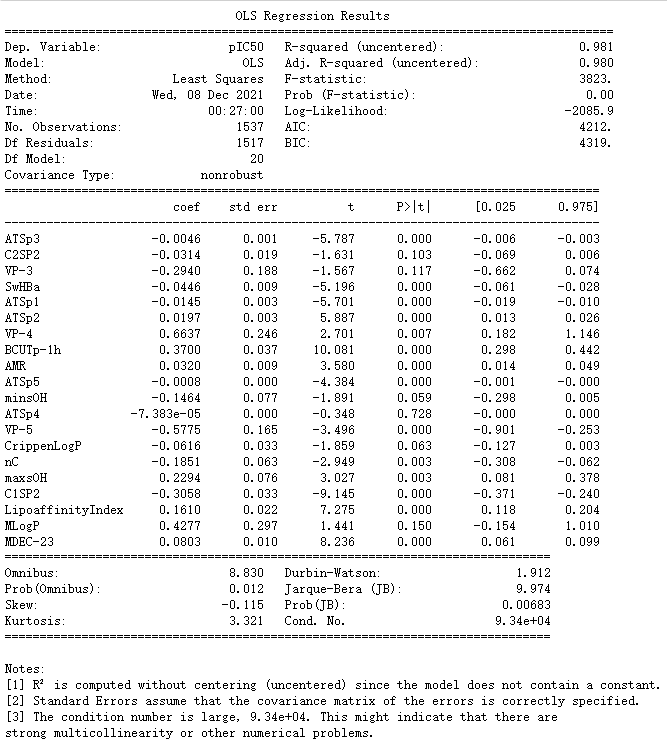


图1.模型概览截图

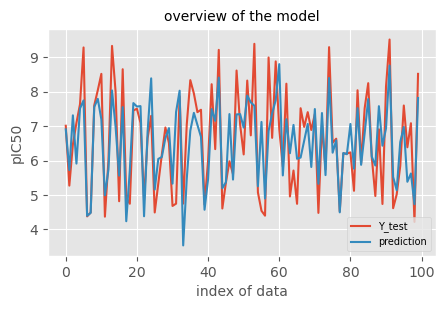


图2.测试集上预测值与真实值对比(取前100条)

**4. 拟合优度计算**

在分割好的测试集上使用模型，得出预测值，计算拟合优度。

1. **import** sklearn.metrics as sm
2. mse2=sm.mean\_squared\_error(Y\_test,results.predict(X\_test))#均方误差
3. rmse2 = np.sqrt(mse)#均方差根
4. evs2=sm.explained\_variance\_score(Y\_test, results.predict(X\_test), sample\_weight=None)#回归方差
5. me2=sm.max\_error(Y\_test,results.predict(X\_test))#最大误差
6. mae2=sm.mean\_absolute\_error(Y\_test,results.predict(X\_test))#平均绝对误差回归损失
7. r22=sm.r2\_score(Y\_test,results.predict(X\_test))#r2检验

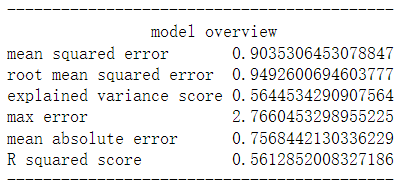


图3.多元线性回归模型拟合优度

**5. 回归系数解读**

根据3.模型概览图中得到的各项参数进行分析：

|  |  |  |
| --- | --- | --- |
| 衡量指标 | 数值 | 描述 |
| R-squared | 0.981 | 决定系数，衡量模型的拟合程度，这里是体现在训练集上的 |
| Adj.R-squared | 0.980 | 相较于R-squared，它会对那些增加的且不会改善模型效果的变量增加一个惩罚项，更适用于多变量线性回归 |
| F-statistic | 3823 | 模型的均方误差除以残差的均方误差，衡量拟合程度。 |
| Prob(F-statistic) | 0.00 | 评估所有变量的显著性水平。零假设是"所有回归系数都为零"。Prob(F-statistics)描述了零假设为真的概率。由以上结果可知，概率接近于零意味着总的来说，回归是有意义的。 |
| Log-likelihood | -2085.9 | 最大似然估计 |
| AIC | 4212 | 衡量统计模型拟合优良性质。在一组可供选择的模型中，通常选择AIC最小的模型。AIC不仅要提高模型拟合度，而且引入了惩罚项，有助于降低过拟合的可能性。  AIC=2k-2ln(L) |
| BIC | 4319 | 与AIC类似,BIC=kln(n)-2ln(L)，惩罚项比AIC大，可以有效避免维度灾难线性。 |
| Prob(Omnibus) | 0.012 | OLS的一个假设是误差是正态分布的。执行综合测试是为了检查这一点。零假设是误差是正态分布的。为了满足OLS假设，我们要求Prob(Omnibus)接近1。在这种情况下，Prob(Omnibus)是0.012，这意味着OLS假设不满足。由于这一点，估计出来的系数不是最佳线性无偏估计(BLUE)。 |
| Durbin-watson | 1.912 | OLS的另一个假设是同方差。这意味着误差的方差是不变的。取值范围为1 ~ 2。在这里，它是1.912，意味着回归结果是可靠的。 |

表2.多元线性回归衡量指标分析

从回归系数上来看ATSp2,VP-4,BCUTp-1h,AMR,maxsOH,LipoaffinityIndex,MlogP,MDEC-23与pIC50成正相关，其他特征与pIC50成负相关。

t统计量是衡量特征显著性的重要指标，P值指的是是当原假设为真时所得到的样本观察结果或更极端结果出现的概率。P值越小，一般以0.05为界，拒绝原来假设的理由就越充分。从模型概览图可得，ATSp4远大于0.05，统计学差异不明显，对回归方程几乎无影响；C2SP2、VP-3、MlogP的p值均在0.1附近，对回归方程影响较小；其余特征都对回归方程有显著影响。

std err，估计系数的基本标准误差，反映的是个系数的精度水平，数值越低，精度越高。观察各个特征的std err可得，大多数系数的std err都在0.1以下，MlogP、VP-5、VP-4、VP-3的std err较高。

1. **问题3**

**1. 构建分类模型**

使用问题一中筛选出的20个特征，并引入本题新引入的五个特征Caco-2、CYP3A4、hERG、HOB、MN，这里选取的是CYP3A4，按照4：1划分训练集和测试集。

* 1. attributes = corr.get\_attributes()  # 在问题一中筛选出的特征
  2. new\_attributes=['Caco-2','CYP3A4','hERG','HOB','MN']
  3. attributes = np.append(attributes,np.array(new\_attributes[1]))#CYP3A4
  4. f1\_attr20 = f1[attributes]
  5. f1\_attr20 = shuffle(f1\_attr20)
  6. X\_train,X\_test,Y\_train,Y\_test = train\_test\_split(f1\_attr20[attributes[:20]],f1\_attr20['pIC50'],train\_size=0.80)

构建逻辑回归模型，默认使用最大似然估计法：

1. **import** sklearn.metrics as sm
2. **import** statsmodels.api as sa
3. model2=sa.Logit(Y\_train,X\_train)
4. result=model2.fit()
5. **print**(result.summary())

在测试集上进行测试：

1. predictedValue=model2.fit().predict(X\_test)
2. p2=predictedValue.copy(deep=True)
3. p2.loc[predictedValue>=0.5]=1
4. p2.loc[predictedValue<0.5]=0

**2. 在测试集上绘制ROC曲线**

1. **from** sklearn.metrics **import** roc\_curve
2. **from** sklearn.metrics **import** roc\_auc\_score
3. fpr2, tpr2, \_ = metrics.roc\_curve(Y\_test,predictedValue)
4. auc2 = metrics.roc\_auc\_score(Y\_test,predictedValue)
5. plt.plot(fpr2,tpr2,label="auc="+str(auc2))
6. plt.legend(loc=4)
7. plt.show()

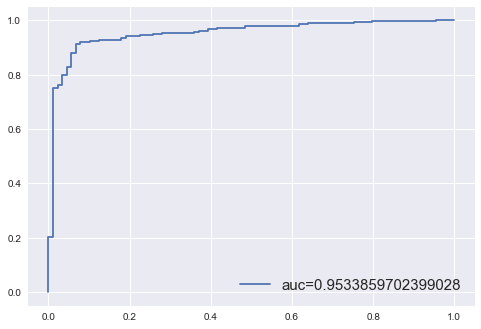
****

图4.ROC曲线图

**3. 解读模型**

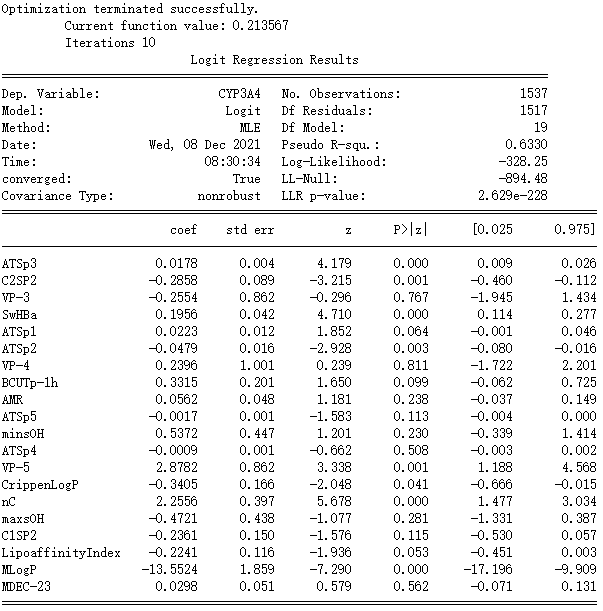
****

图5.逻辑回归模型概览截图

|  |  |  |
| --- | --- | --- |
| 衡量指标 | 数值 | 描述 |
| Pseudo R-squared | 0.6330 | 类似于最小二乘法下的R-squared，计算公式为(lnL0-lnL1)/lnL0,lnL1为原模型的对数似然函数最大值，lnL0为以常数项为唯一解释变量的对数似然函数的最大值。用于衡量模型的优度。 |
| Log-Likelihood | -328.25 | 对数似然函数的最大值 |
| LL-Null | -894.48 | 当只包含一个截距时最大对数似然函数的结果，是计算pseudo R-squared和Log-Likelihood Ratio的基础 |
| LLR p-value | 2.6296e-228 | LL-Null和LLR p-value相当于线性回归的F-statistic和F-proba。评估所有变量的显著性水平。 |

表3.逻辑回归模型衡量指标

CYP3A4：‘1’代表该化合物能够被CYP3A4代谢，‘0’表示该化合物不能被CYP3A4代谢。z值与p值衡量了特征的显著性水平，从模型概览截图中我们可以看出，VP-3、VP-4、AMR、VP-5、maxsOH、CrippenLogP、ATSp5、ATSp4、MDEC-23等特征的p值大于0.05，这表明统计学差异并不显著，以上几个特征对模型的影响不大。ATSp3、C2SP2、SwHBa、ATSp2、VP-5、nC、MlogP等特征的p值都小于0.01，表明统计学差异显著，对模型有着较大影响。

std err是系数的基本标准误差，从模型概览截图中可以看出，MLogP、nC、VP-3、VP-4、minsOH、VP-5、nC的std err较大，均在0.5以上。ATSp3、ATSp1、ATSp5、ATSp4的std err较小。

**Odds ratio计算：**

1. odds\_ratio=np.exp(result.params)
2. **print**(odds\_ratio)

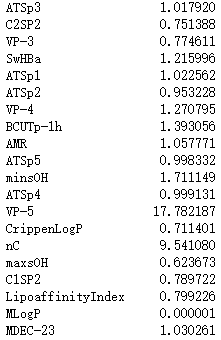


图5. Odds ratio 结果

Odds Ratio反映了特征对模型的影响程度，OR值大于一，代表特征对模型有着较强影响，OR值小于或等于一，代表特征对模型影响较小或几乎无影响。

**4. 质量评估**

1. **from** sklearn.metrics **import** classification\_report
2. **print**(classification\_report(Y\_test,p2))

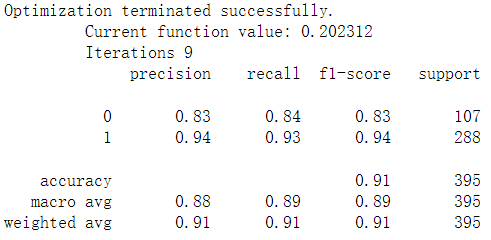


图6.质量评估截图

|  |  |
| --- | --- |
| 评估指标 | 描述 |
| recall | 分类模型找到所有正面实例的能力。对于每个类别，它被定义为真阳性与真阳性和假阴性之和的比率。 |
| precision | 分类器不把实际是负面的实例标记为正面的能力。对于每个类，它被定义为真阳性与真阳性和假阳性之和的比率。 |
| f1-score | recall和precision的加权调和平均值，最好的分数是1.0，最差的分数是0.0。F1得分低于精度测量，因为它们将precision和recall嵌入到计算中。一般而言，应该使用F1的加权平均值来比较分类器模型，而不是全局精度。 |
| support | 类别在数据集中实际出现的次数。训练数据中的不平衡support可能表明分类模型报告的分数的结构弱点，并可能表明需要分层抽样或再平衡。support不改变模型，而是诊断评估过程。 |
| AUC | 已经在2.中计算过，衡量分类器区分类别能力的指标，被用作ROC曲线的总结。AUC越接近1，模型越完美 |

表4.逻辑回归评价指标