AutoCash API文档：

**执行、修改代码的一些注意事项：**

**1.若需要减少元特征，只需要在Features里面删除features\_calculation里面的项即可**

**2.ClassSelectionMain.py中给出的需要训练的示例中所给时间基本可以视为程序不报错的最少时间，在该时间下运行效果很差，完全训练大约需要用时10h**

**3.改动算法时需要在WekaClassifier.txt与HPO.txt中同时修改，注意格式一致**

1. 数据集加载文件及输入格式说明

**ClassSelectionMain.py**

输入：

:param filename:数据集路径

:param alg\_selection\_time:程序最大运行时间(s)

:param hpo\_iteration\_num:hpo中遗传算法优化代数

:param evaluation\_indicator:AUC/ACC/……

:param classp:标志数据集分类信息是在最前一行或是最后一行(start/end)

:param train\_flag:是否重新进行dqn训练(y/n)

输出：最优方案（Dict【算法名称：OptAlgName；超参数设置：Dict【超参数名称：超参数值】】）

**loadMetaFeature.py:**

loadUserData(filename,classp):加载用户输入数据集的信息

输入要求：

filename： str

内容：用户数据集的url，格式要求：.arff

Classp：str

内容：用户数据集中的分类信息是在第一列还是最后一列，分别对应“start”和“end”

输出：绘制一张饼图，可在当前文件夹下的“pie.png”中查看

**CaculateFeatures.py**

loadData(featurelist, filename, classp):

输入要求：

Featurelist：list

内容：元特征索引，即上述dqn\_train的返回值

filename： str

内容：用户数据集的url，格式要求：.arff

Classp：str

内容：用户数据集中的分类信息是在第一列还是最后一列，分别对应“start”和“end”

输出：需要训练的数据集的元特征信息（具体数据）

**GenerateDataSet.py:**

GenerateDataSetaction(FeatureList):计算训练数据集的元特征表示

输入要求：

FeatureList：list

内容：元特征索引，即上述dqn\_train的返回值

输出：如果不进行元特征训练，输出内容可在当前文件夹下的dataset.txt中查看，若进行元特征训练，输出内容可在dataset1.txt中查看

1. 算法列表代码说明

**GenerateData.py:**

Allocationalgorithm(time\_limit,evaluation\_indicator):为训练数据集分配算法

输入要求：

time\_limit：时间限制

evaluation\_indicator：评估指标名称

输出：输出每个训练数据集的算法分配情况，可在Customization\_result.txt中查看

注：所有可选择的算法的列表可以在WekaClassifiers.txt文件中查看，如需要增删算法，请在WekaClassifiers.txt中按格式增加，同时注意两次增加需要确保两个算法在两处的索引相同。同时注意在HPO.py文件中进行相应增删

1. 特征提取代码说明

**Dqn.py:**

dqn\_train(dqn\_time\_limit):dqn训练

输入要求：

dqn\_time\_limit：int

内容:dqn训练的最长时限，超过此时限暂停运行dqn，输出最后一次结果，单位：秒。注：如果不设时间限制dqn训练大约需要3-4小时

输出：输出选择的元特征索引的list形式，对应的具体名称可在main.py中的features\_calculation查看。也可以在当前文件夹下的Selected\_features.txt查看

注：默认元特征列表为FeatureList=[0, 2, 4, 6, 7, 9, 13]

如果需要增加或删除元特征，可在CalculateFeatures.py中增加元特征的计算方法，并注意保持FeatureList的索引与具体数值之间的一一对应关系。

1. 算法选择代码说明

**Mymodel.py**

modelFit(X\_train,FeatureList,data\_file):推荐用户数据集适合的算法

输入要求：

X\_train：list

内容：需要算法推荐的数据集的元特征表示，使用loadData(featurelist, filename, classp)的返回值

输出：str

为该数据集推荐的算法，格式与WekaClassifiers.txt中的格式相同，如需要输入hpo进行超参数优化注意格式转换，可见main.py54行alg\_name处理方式

1. 超参数优化代码说明

**HPO.py:**

hpo(HPO\_name,filename,hpo\_iteration\_num):

输入要求：

HPO\_name:str

内容：算法推荐后输出的算法名，注意：格式如alg\_name,例如可以输入："AdaBoostM1"

Filename：

内容：需要超参数优化的数据集的url

hpo\_iteration\_num：

内容：最大优化代数

输出：优化后的超参数与超参数的值

1. Evaluation，算法评估

Evaluation.py:

evaluation(filename,evaluation\_indicator,classifier,option,classp):

输入内容:

:param filename:数据集路径

:param evaluation\_indicator: AUC/ACC/……

:param classifier: 算法名称

:param option: 超参数值列表

:param classp: 标志数据集分类信息是在最前一行或是最后一行(start/end)输出：list

内容：算法在给定数据集下及超参数设置下的性能值，即评估指标数据（Float）

1. 请单独为整个方法支持的算法列表和数据特征列表进行总结，各列一个表格。

算法列表：均为weka中的算法，具体名称与超参数如下

|  |  |
| --- | --- |
| 算法名称 | 超参数 |
| weka.classifiers.meta.AdaBoostM1 | ["-P","-I","-Q"] |
| weka.classifiers.bayes.BayesNet | ["-D"] |
| weka.classifiers.meta.Bagging | ["-I","-P","-O"] |
| weka.classifiers.meta.ClassificationViaRegression | ["-M"] |
| weka.classifiers.rules.DecisionTable | ["-X"] |
| weka.classifiers.lazy.IBk | ["-E","-X","-I","-K"] |
| weka.classifiers.lazy.KStar | ["-M","-B"] |
| weka.classifiers.functions.Logistic | ["-R"] |
| weka.classifiers.meta.LogitBoost | ["-O","-P","-H"] |
| weka.classifiers.functions.MultilayerPerceptron | ["-L","-M","-B","-C","-D"] |
| weka.classifiers.bayes.NaiveBayes | ["-D","-K"] |
| weka.classifiers.meta.RandomCommittee | ["-I"] |
| weka.classifiers.trees.RandomForest | ["-I","-depth","-K"] |
| weka.classifiers.meta.RandomSubSpace | ["-I","-P"] |
| weka.classifiers.meta.Vote | ["-S"] |
| weka.classifiers.rules.JRip | ["-F", "-N", "-O", "-S"] |
| weka.classifiers.trees.LMT | ["-I", "-M", "-W"] |

数据特征列表：支持23个元特征，现将名字列出：

1.目标属性类别个数

2.目标属性类别的信息熵

3.目标属性单个类别占的最大比例

4.目标属性单个类别占的最小比例

5. numeral 属性个数

6.category 属性个数

7.numeral 属性所占比例

8.属性总数

9.record 个数

10.category 属性中类别最少的属性 的类别个数

11.category 属性中类别最少的属性 的类别的 信息熵

12.category 属性中类别最少的属性 的类别中 单个类别占的最大比例

13. category 属性中类别最少的属性 的类别中 单个类别占的最小比例14.category 属性中类别最多的属性 的类别个数

15.category 属性中类别最多的属性 的类别的 信息熵

16.category 属性中类别最多的属性 的类别中 单个类别占的最大比例17.category 属性中类别最多的属性 的类别中 单个类别占的最小比例

18.numeral 属性中最小的平均值

19.numeral 属性中最大的平均值

20.numeral 属性中最小的方差

21.numeral 属性中最大的方差

22.numeral 属性的平均值的方差

23.numeral 属性的方差的方差