# 1、随机森林算法（RandomForest）

|  |  |
| --- | --- |
| 所在包 | weka.classifiers.trees.RandomForest |
| 默认参数 | -I/50/-K/8/-S/12/-depth/0/-B/-num-slots/4 |
| 映射后示例 | I/50/K/8/S/12/H/0/B/N/4 |

参数说明：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数名称 | 映射 | 参数解释 | 默认 | 例子 | 备注 |
| -I | I | 组成森林的树的棵树 |  | I/100 | 棵树太少效果不佳，太多训练太慢，模型也会太大，一般选20-500足够 |
| -K | K | 构建单棵树随机选取的特征数目 |  | K/10 | 每棵树都取部分特征，一般考虑原特征个数的平方根+1 |
| -depth | H | 单棵树的最大深度 | 0 | H/10 | 如果指定为0则不限制深度，可能会导致过拟合。可以先指定为0看效果，如果recall或precision太低可以试着降低树深度 |
| -num-slots | N | 并行时使用的CPU核数 | 1 | N/4 | 1表示不并行。请根据机器的CPU核心数调整，指定多核可提高训练速度，核心数太多可能会导致CPU负荷过重 |
| -D | D | Debug模式执行 |  | D | 输出更多详细信息 |
| -S | S | 随机数种子 | 1 | S/10 | 可能对最终结果略有影响 |

# 2、SVM

|  |  |
| --- | --- |
| 所在包 | weka.classifiers.functions.LibSVM |
| 默认参数 | -K/2/-G/1/-C/20/-W/1 25/-M/500/-Z |
| 映射后参数 | K/2/G/1/C/20/W/1 25/M/500/Z |

参数说明：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数名称 | 映射 | 参数解释 | 默认值 | 例子 | 备注 |
| -S | S | SVM的类型 | 0 | S/0 | 0 = C-SVC  1 = nu-SVC  2 = one-class SVM  3 = epsilon-SVR  4 = nu-SVR  用于分类，选择0就可以了 |
| -K | K | 核函数类型 | 2 | K/2 | 0 = linear: u'\*v  1 = polynomial: (gamma\*u'\*v + coef0)^degree  2 = radial basis function: exp(-gamma\*|u-v|^2)  3 = sigmoid: tanh  (gamma\*u'\*v + coef0)  维度很高可以尝试1线性核，一般默认2高斯核 |
| -D | D | 多项式核最高次项的次数 | 3 | D/3 | 使用此参数前提是 – K参数为1，一般选择1-11 |
| -G | G | 核函数的gamma参数 | 1/m | G/0.5 | 使用此参数前提是-K=2,m是分类问题的类别数目，对于高斯核函数（-K/2），通常可选择下面几个数的倒数：0.1，0.3，0.8，1.6，3.2，6.4 ，12.8 |
| -R | R | 多项式核和sigmoid核的coef0  参数 | 0 | R/0.5 | 一般选0.2，0.4，0.6，0.8，1 |
| -C | C | C-SVC, epsilon-SVR, nu-SVR中损失函数惩罚系数 | 1 | C/10 | 对分类的效果影响非常大，可以使用Grid-search和Cross-validation筛选，可以从0.0001 到10000，选择的越大，表示对错误例惩罚程度越大，可能会导致模型过拟合，也可能增加使模型学习时间。一般来讲，C取10-100比较合适，recall过低可以尝试增加C，precision过低可以尝试降低C，反之亦然。 |
| -N | N | nu-SVC, one-class SVM, nu-SVR中 | 0.5 | N/1 | 使用one-class SVM时指定此参数 |
| -Z | Z | 开启数据的标准化 | Off | Z | 将数据按比例缩放，使之落入一个小的特定区间。将数据转化为无量纲的纯数值，便于不同单位或量级的指标能够进行比较和加权，除非原始数据已经进行过标准化，否则必须开启。 |
| -J | J | 关闭数据的nominal to binary  转化 | On | J | 将名义型变量进行编码，使其转化为若干列0,1两种状态的编码，避免存在大小关系。若输入数据没有名义型变量可关闭此功能，否则必须开启。 |
| -V | V | 关闭数据的缺失值处理 | On | V | 自动处理缺失值，如果数据没有缺失值可以关闭此功能。 |
| -M | M | 设置cache大小，单位是MB | 40 | M/100 | 设置较大内存可以加快训练速度，内存比较大的机器可以设置较大的cache |
| -E | E | 设置终止迭的最小误差 | 0.001 | E/0.1 | 设置的太小可能导致学习时间过长，一般默认就可以了 |
| -H | H | 关闭shrinking | On | H | 关闭可能会提高训练速度，若训练太慢，可以尝试关闭 |
| -W | W | 第i个类的权重 | 均为1 | W/ 1 20 | 目标类有几个就写几个权重，由空格分隔，如果想把某一类的权重提高（比如该类比较少，更关注该类），可以给该类设置一个较大的权重，如果某一类样本的recall比较低， |
| -B | B | 输出每个类的概率估计 |  | B | 输出概率值，相当于可信度评分 |
| -seed | S | 随机数种子 | 1 | S/2 | 可能影响最后效果，可以尝试更改 |

# 3、NaiveBayes

|  |  |
| --- | --- |
| 所在包 | weka.classifiers.bayes.NaiveBayes |
| 默认参数 | 无 |
| 备注 | 无 |

参数说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名称 | 参数解释 | 默认值 | 例子 | 备注 |
| -K | 对于连续型变量，使用核密度估计代替高斯函数估计 | K | -K | 效果不佳可以尝试 |
| -D | 对于连续型变量进行有监督离散化 | D | -D | 如果原始数据存在连续型变量，最好开启离散化，可能影响最终效果。 |

# 4、logisticRegression

|  |  |
| --- | --- |
| 所在包 | weka.classifiers.functions.Logistic |
| 默认参数 | -R/0.01/-M/1000 |
| 映射后参数 | R/0.01/M/1000 |

参数说明：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数名称 | 映射 | 参数解释 | 默认值 | 例子 | 备注 |
| -R | R | L2正则化参数 |  | R/0.1 | 设置L2正则的系数，可以改善过拟合程度，如果模型的recall或precision过低，可以调节该参数 |
| -M | M | 最大迭代次数 | -1  直到收敛 | M | 建议指定一个最大迭代次数，否则可能会无法收敛，模型将一直训练 |

# 5、CART

|  |  |
| --- | --- |
| 所在包 | weka.classifiers.trees.J48 |
| 默认参数 | -M/5/-R/-B/-L |
| 备注 | M/5/R/B/L |

参数说明：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数名称 | 映射 | 参数解释 | 默认值 | 例子 | 备注 |
| -U | U | 使用不剪枝树 | off | U | 使用不剪枝的树，使用该参数后树将任意生长，可能导致过拟合。如果作为弱模型使用，不推荐使用此参数，如果作为强模型使用，要防止过拟合。 |
| -C | C | 设置剪枝阈值 | 0.25 | C/0.1 | 阈值越高，可能发生的剪枝越多，如果模型过拟合（在测试集上表现不佳），可以尝试提高阈值 |
| -M | M | 每个叶子上的数据条数 | 2 | M/3 | 每个叶子的数据条数太少，模型过拟合的风险将会增大，如果模型过拟合，尝试提高每个叶子上的数据条数 |
| -R | R | REP剪枝策略 | off | R | 使用reduced error pruning剪枝策略，即在满足剪枝条件时，先执行剪枝操作，当剪枝确实降低了错误率，才真正执行剪枝，否则不剪枝，推荐使用 |
| -N | N | 设置REP剪枝策略时的fold数目 | 3 | N/5 | 一个fold用于验证剪枝效果，其余的用于树的生长，当制定了-R参数时使用 |
| -B | B | 树的每个节点只使用二分裂 | off | B | 该参数使用后，树的每个节点将智能进行二元分裂，最后构成的树将是二叉树，推荐使用 |
| -S | S | 关闭子树提升 | on | S | 子树提升策略会将子树与他上一级树进行替换来比较效果，不推荐关闭。 |
| -A | A | 使用拉普拉斯平滑 | off | A | 推荐使用 |
| -J | J | 关闭连续属性离散化 | on | J | 有监督的离散化连续性数值数据,如果数据中存在连续值，必须使用。 |

# 6、Adaboost

|  |  |
| --- | --- |
| 所在包 | weka.classifiers.meta.AdaBoostM1 |
| 默认参数 | -S/1/-I/50/-W/weka.classifiers.trees.J48/--/-U/-B/-L |
| 映射后示例 | S/1/I/50/W/CART/#/U/B/L |

参数说明：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数名称 | 映射 | 参数解释 | 默认值 | 例子 | 备注 |
| -P | P | 前级训练使用的数据比例 | 100  全量 | P/80 | 一般指定90，使用部分数据可以提高速度 |
| -S | S | 随机数种子 | 1 | S/2 |  |
| -I | I | 迭代次数 | 10 | I/20 | 次数太大会降低训练速度，一般没有必要指定太大，如果效果不好可以尝试提高迭代次数 |
| -W | W | 要提升的模型的完整名字 | weka.classifiers.  trees.DecisionStump | W/  LR | 这里的模型一定要是弱分类模型，也就是logist/naivebayes/CART/SVM等，不包括randomforest |
| -- | # | 要提升模型的参数 |  |  | 基础模型的参数附在--/后面，格式与之前基本模型参数设置方法一致 |

# 7、Stacking

|  |  |
| --- | --- |
| 所在包 | weka.classifiers.meta.Stacking |
| 默认参数 | -X/4/-M/weka.classifiers.functions.Logistic |
| 备注 | X/4/M/LR |

参数说明：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数名称 | 映射 | 参数解释 | 默认值 | 例子 | 备注 |
| -X | X | 交叉验证的fold数 |  | -X/10 | 一般范围是3-10，次数越大，训练时间越长 |
| -S | S | 随机数种子 | 1 | -S/2 |  |
| -M | M | 第二级模型的完整名称 |  | -M/ weka.classifiers  .functions.Logistic | 一般选择Logistic即可，也可以尝试其他模型 |

# 8、GBDT

|  |  |
| --- | --- |
| 参数 | -eta/0.2/-max\_depth/10/-scale\_pos\_weight/5/-num\_round/7 |
| 映射后示例 | E/0.2/H/10/W/5/N/7 |

参数说明：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数名称 | 映射 | 参数解释 | 默认值 | 例子 | 备注 |
| -eta | E | 在提升过程中用于对抗过拟合的收缩参数可以，可以收缩提升过程中的特征向量权重，使更新更为保守 | 0.3 | E/0.3 | 取值范围[0,1] |
| -gamma | G | 进行叶子节点划分的最小损失下降阈值，误差减少少于此值就不继续分裂，显然越大算法表现越保守 | 0 | G/2 | 取值范围：[0, ∞] |
| -max\_depth | H | 树的最大深度 | 6 | H/10 | 取值范围：[1, ∞] |
| -min\_child\_weight | S | 分裂节点权重和的最小值，分裂后孩子上的样本权重和低于该阈值则取消本次分裂 | 1 | S/1 | 取值范围：[0, ∞] |
| --max\_delta\_step | D | 每颗树所能贡献的最大预测权重。 | 0 | D/2 | 可以帮助控制提升过程更保守，建议设置在[1,10]之间。为0，代表没约束。取值范围：[0, ∞] |
| -lambda | L | L2正则项的权重 | 1 | L/1 |  |
| -alpha | A | L1正则项的权重 | 0 | A/0 |  |
| -scale\_pos\_weight | W | 控制正负样本权重的平衡 | 1 | W/5 | 可以设置 负样本数量/正样本数量。 |
| -refresh\_leaf | R | 在更新节点时是否更新叶子节点 | 1 | R/1 | 为1时更新叶子节点。  为0时不更新叶子节点。 |
| -num\_round | N | 提升算法的计算次数 | 5 | N/5 |  |

# 界面交互映射说明：

|  |  |  |
| --- | --- | --- |
| 名称 | 完整包名 | 映射值 |
| 决策树 | weka.classifiers.trees.J48 | CART |
| SVM | weka.classifiers.functions.LibSVM | SVM |
| 朴素贝叶斯 | weka.classifiers.bayes.NaiveBayes | NB |
| Logistic回归 | weka.classifiers.functions.Logistic | LR |
| 随机森林 | weka.classifiers.trees.RandomForest | RF |