**不同水文环境下船模能否出坞下水的基于规则的决策模型**

1. 模型用途

给定降水、蒸发和径流等水文循环等水文要素的情况下，判断不同水文条件下船模能否出坞下水。

1. 数据说明

输入数据用变量表示，共有个特征，表示配送过程中各类环境因子的历史情况：

第一个特征表示：船模出坞下水时的降水情况 (mm/h),

第二个特征表示：船模出坞下水时的蒸发情况 (mm/h),

第三个特征表示：船模出坞下水河道的径流情况 (m3/s),

第四个特征表示：船模出坞下水河道的水位 (m),

第五个特征表示：船模出坞下水河道的流速 (m/s),

第六个特征表示：船模出坞下水河道的结冰情况,

第七个特征表示：船模出坞下水河道的含沙量 (kg/m3),

输出数据用y表示，只有1个特征：

船模能否出坞下水，如果能则，否则.

1. 算法原理

不同水文环境下船模能否出坞下水的基于规则的决策模型的核心思想是根据从训练数据中训练而来的决策规则，对进行预测。一条决策规则就是一个简单的IF-THEN语句，由条件（也称为先行条件）和预测组成。 比如我们现在要使用一种算法来学习判断不同水文下船模能否出坞下水（能或不能）的决策规则。该模型学得的一个决策规则可能是：如果水位大于100米并且流速小于90km/h时，则可以出坞下水。更正式地讲：IF depth>100m AND velocity<90km/h ，THEN value=1。

让我们分解一下决策规则：

• depth>100m 是IF部分的第一个条件。

• velocity<90km/h是IF部分的第二个条件。

• 将这两个条件用AND相关联创建新条件。意即两者都必须为真，才能应用规则。

• 预测的结果（THEN部分）为value=1。

决策规则在条件中至少使用一个feature = value语句，并且对使用AND的数量没有上限。 单个决策规则或几个规则的组合可用于进行预测。

1. 数据预处理

由于决策树模型对数据特征的尺度不敏感，因此除了填补数据缺失值外，不需要特别进行数据归一化或者去量纲等预处理步骤。

（二）数学建模

（a）算法步骤

决策树模型是极具可解释性的预测模型，一个典型的代表是ID3算法。 ID3 算法遵循奥卡姆剃刀原理：用小型简单的决策树构建效果尽可能好的模型。

信息论中信息熵越大，样本纯度越低。ID3 算法的核心思想就是以信息增益来度量特征选择，选择信息增益最大的特征进行分裂。算法采用自顶向下的贪婪搜索遍历可能的决策树空间。 其大致步骤为：

1. 初始化特征集合和数据集合；

2. 计算数据集合信息熵和所有特征的条件熵，选择信息增益最大的特征作为当前决策节点；

3. 更新数据集合和特征集合（删除上一步使用的特征，并按照特征值来划分不同分支的数据集合）；

4. 重复 2，3 两步，若子集值包含单一特征，则为分支叶子节点。

（b）特征选择标准

ID3 使用的分类标准是信息增益，它表示得知特征 A 的信息而使得样本集合不确定性减少的程度。

假设数据集上有 个类 ， 为此类样本个数，显然有 ，等号右边是数据集样本个数。假设特征 有且仅有 个可能的取值，并且对应不同取值将数据集划分为 个子集 ，则显然又有 。有了这些记号，我们可以计算两个重要的量，经验熵和经验条件熵，以及由这两者之差而得到的信息增益，一般情形下又叫做叫做互信息。

数据集 按类别概率的经验熵 ，

特征 对数据集 类别的经验条件经验熵 ，

上式其实就是在各属性划分的子集 上计算条计算经验熵。然后根据这个子集样本的比例作为这些经验熵的权重累和。而子集 上计算条计算经验熵和在 上计算经验熵一样，这时把 看做整个数据集。最终我们得到信息增益：

1. 使用方法

运行model.py文件得到判断不同水文条件下船模能否出坞下水，具体输入命令：

**python chuanwuDT.py --trfile 训练数据 --tefile 测试数据 -o 结果文件名**

例如：

**python chuanwuDT.py --trfile ./environmentDat.txt --tefile ./environmentTestDat.txt -o ./result.txt**

预测结果会存放在**./result.txt**文件中