决策树迷你项目

在这个项目中，我们使用决策树再次将邮件进行分类，启动代码是 `decision_tree/dt_author_id.py`。

第一部分：让决策树运行起来

构建一个决策树运行的分类器，设置 `min_samples_split=40`。开始训练之前，这可能需要一些时间。那精确度是多少？

第二部分：加速

在 SVM 迷你项目中，你会发现参数调节能显著加速机器学习算法的训练时间。一般的规律是参数可以调整算法的复杂度，通常更加复杂的算法意味着运行得更慢。

另外一个方法是通过训练/测试所使用的特征数量控制算法的复杂度。算法中可用的特征越多，出现复杂拟合的可能性就越大。我们会在“特征选择”的课程中详细讨论这个问题，不过现在你可以先预览一下。

- 从你的数据中找出特征的数量，数据是以 numpy 数组的形式排列的，其中数组的行数代表数据点的数量，列数代表特征的数量；为了码提取这个数值，可以写一行这样的代码 `len(features_train[0])`
- 加入 `tools/email_preprocess.py`，会看到这样的代码：`selector = SelectPercentile(f_classif, percentile=1)`，将 `percentile` 从 10 改为 1。
- 现在的特征数量是多少呢？
- 你认为 SelectPercentile 起到什么作用？其他所有的都不变的情况下，赋予 percentile 的值较大是否得到一棵更加复杂的或者简化的决策树？
- 注意训练时间的不同取决于特征的数量。
- 当 percentile 等于 1 时，准确度是多少？

翻译：Iris
2016 年 9 月