项目文件	作用
base_eva.py	获取 air 或 mete 特征，直接用分类器分类，如 SVC。 在代码里可以选择单独使用 air，mete 特征或合并使用。
consistent_eav.py	对于每个测试样本，选取与其相关度较高的训练样本用于预测
SGD_weight_eva.py	原理同 consistent_eva.py，但是： （1） 分类器为 SGDClassifier，对于每个测试样本，用每个训练样本与它的相关度作为该训练样本的权重 （2） 有不同的过滤条件（4 种不同的函数，不同的阀值）
txt2csv.py 参数： (train/test/total): 读取 train 样本还是 test 样本还是都读进来合并 m 二元分类的第一类 n 二元分类的第二类（0,1,2,3,4,5） (1/2) 是单独使用 air 还是结合 air, mete (single/all), 二元分类（只读取 m,n 两类的数据）还是读取所有类别的数据	在使用 moa 的 HoeffdingAdaptiveTree 和 OzaBagAdwin 时，采用的是 one-against-one 的策略。为此需要先得出每个 binary 分类器的精确度。此文件的作用是根据参数提取出每个二元分类器所需要的数据并存为 csv 格式，便于后续直接用 weka 提供的接口转为 arff 文件。
weka.jar	用里面的 weka.core.converters.CSVLoader 接口将 csv 转成 arff
ridofcomma.py 参数：作用同 txt2csv.py Train/test/total M n	对上述生成的 arff 文件做一些调整，例如要分类的类别等。
moa.jar sizeofag.jar	Moa 在命令行中使用
multi_accuracy.py	根据 one-against-one 策略合并所有上述二元分类器对全部测试数据的分类结果，并得到最终合并的预测结果，统计精确度
Concept_drift_multiaccuracy.bat 可以在代码里控制运行 4 部分里的那一部分 实际上只有第 4 部分是 concept drift，前面三部分都是最初用 evaluatemodle 而不是 evaluateprequential(合并训练和测试数据,concept drift)时采用 one-against-one 方法时的工作。 最后在做实验时是使用 moa 的 evaluatePrequential 方法做 concept drift，所以之前的二元分类这部分可以说作用不大。	用批处理脚本文成上述所有二元分类和最终合并的处理。 具体分成 4 部分： 1. 读取所有的 test sample 并转成 arff 文件 因为每个二元分类器都要对全部测试样本进行预测 读取每组二元分类的训练样本并进行训练，再运用得出的分类器对全部测试样本进行预测 2. 运用 multi-accuracy.py 得到合并后多元分类的结果（需要 1 的运行结果） 3. 单独统计每个二元分类器用于自己对应

	的两类数据预测时的精度 4. 合并 train sample 和 test sample，全部作为训练数据，运用 moa 的 EvaluatePrequential 方法(concept drift)进行预测
arima.r	R 语言实现的 arima 模型预测
splitweight.py	在比较不同的 correlation(0,1,2,3,01,02…)时，用 job3 读取每对 snapshot_i 和 snapshot_j 所对应的不同种类的 correlation 值时，是把这 15 种情况的值合并成 1 个字符串并用#分割。此文件用于解析上述读的结果并为每种 correlation 单独写一个文件。
Correlation_compare.bat	在比较不同的 correrlation 时，需要遍历 15 中 correlation 和从 0.5 到 0.95 的阀值，用此批处理脚本完成
Easy.py	简单的脚本，将上述结果写入 csv 文件
Forecastio 文件夹	爬取数据
Smog_forecast	从数据库读取数据 java 工程

使用方法：

1. 可以直接运行 base_eva.py, consistent.py, SGD_weight_eva.py 测试对某次样本的测试
2. Arima 模型测试直接运行 arima.r
3. 不同过滤函数、阀值可以在 SGD_weight_sva.py 里设置
4. Concept drift 部分可以直接运行 Concept_drift_multiaccuracy.bat 的不同部分完成（结果保存在 concept drift 文件夹里）
5. Correlation 比较部分，先运行 splitweight.py，再运行 correlation_compare.bat 完成（结果保存在 different_correlation 文件夹里）。对于每一种 correlation 和阀值，通过 SGD_weight_eva.py 得到预测精度。

每次样本的意义：

样本文件存在 samples 文件夹中

Exp	意义
006	aqi(t-5)~t
007	aqi(t-5)~t air spatial(t)邻近两个 station
008	Aqi(t-5)~t air
009	同 008
010	同 008
Consistent_weight.txt consistent_train_txt	原先的 correlation
Consistent_weight_10.txt	15 种 correlation 合并保存的文件
total_009.txt	Train samples 和 test samples 合并在一起的文件