数据测试项目

本平台侧重于大数据测试，重在spark平台，测试算法性能、数据预处理都采用提交jar包的方式。目前支持scala、java两种语言，也是spark支持的语言。

算法性能测试

算法性能测试模块在spark平台上运行。由于spark采用scala编写，因此，在这个模块，我们仅支持scala、java两种语言。程序提交后，根据固定流程进行，需要特定的参数，但是由于算法的不一致性，很难对所有算法形成统一框架，为了保证算法的多样性，势必造成操作的混乱，因此在此形成一套适用于算法性能测试的代码规范。

1-1 连接集群

连接集群的 `setMaster()` 默认为用户设置的集群ip地址，因此在算法中无需体现。

1-2 执行类名

所执行的类必须命名为 `Main`

1-3 参数设置

算法采用从外获取参数的方式，在本模块中，前四个参数固定存在，有固定用途，固定格式：

```scala
def main(args: Array[String]): Unit = {
    /*
    * 1.第零个、第一个参数(args(0), args(1)): 分别为 url 与 id,
    *   此处不需要指定值，只需要预留位置即可，此为连接连接后台的接口，具体值在后台指定
    * 2.第二个、第三个参数(args(2), args(3)): 分别为 input 与 output,
    *   input是外部文件输入的路径，output是算法测试后，结果保存的路径
    */
    val url = args(0)
    val id = args(1)
    val input = args(2)
    val output = args(3)

    /*复制下面一行在正式算法开始之前，用于向告知后台程序开始运行*/
    val start = Http(url).postForm.param("id",id).param("status","1").asString

    // ----此处是算法性能测试的开始----
    kmean(input, output)
    // ----此处是算法性能测试的结束----

    /*复制下面一行在正式算法解释之后，用于向告知后台程序运行结束*/
    val end = Http(url).postForm.param("id", id).param("status","0").asString
  }
```

对于有监督算法，需要分训练集(特征与标签)、测试集(只有特征，没有标签)，其中，测试集输入不变 `val input = args(2)`， 测试集输入 `val test_input = args(4)`， 即有监督学习多了一个固定参数。

以上，固定四个(有监督学习中五个)参数的位置与用途，如果算法需要另外加参数，则在代码中 继续添加 `args(4), args(5)` 等即可。

1-4 结果评估（可选）

若算法性能测试后，还需要使用本平台进行结果评估，由于后续需要对算法结果进行评估，因此还需要对算法的输出格式作出要求。(若不需要对结果做评估，则可忽略本条)

目前预置对三种算法结果的处理，分别为**聚类、分类、回归**。

由于技术限制，对spark处理的结果，无法用rdd或者其他格式评估，需要将结果转为最普通的 `.txt` 格式，对于不同的算法，其结果中包含的内容也有所差别：

1. 对聚类算法：

   由于后续对聚类算法的评估需要用到聚类中心点，因此聚类中心点坐标需要保存在最后结果中，以 `=` 结尾，用以区分聚类中心点和其他点。

   其他点则不需要特殊处理。满足**使用英文 `,` 作为同一行的数字之间的分隔，不应包含其他信息，例如：数组格式外的 `[`, `]`,空格等。**

   例如下图：前三行代表聚类中心的表示，用 `=` 结尾，其他点紧随其后，不作改变。

   ```
   5.005999999999999,3.4180000000006,1.464000000000002,0.243999999999=
   5.901612903225806,2.74838707741932,4.393548096774,1.433870967741935=
   6.85,3.0736842105263147,5.742105263157893,2.071052631578947=
   5.1,3.5,1.4,0.2,0
   4.9,3.0,1.4,0.2,0
   4.7,3.2,1.3,0.2,0
   ...
   ```

2. 对分类、回归算法：

   对于回归、分类等有监督算法，**需要将预测结果放置最后一列**，同时满足**使用英文 `,` 作为同一行的数字之间的分隔，不应包含其他信息，例如：数组格式外的 `[`, `]`,空格等。**

> 凡是在本平台测试的算法性能的都需要遵循第1-1、1-2、1-3条。
>
> 若对算法结果有自己的评估方式，或者不需要对算法结果进行评估，则不需要遵循1-4条，若需要在本平台评估算法结果(仅包括预置的聚类、分类、回归三种算法)，则还需遵守第1-4条。

数据预处理过程

考虑到数据预处理的一般性，在这个过程中，仅支持python对数据进行预处理。

同算法性能测试一样，数据预处理的提交过程也一样固定了四个参数，固定用途。

```
if __name__ == '__main__':
    # python的外部从第1个起，作为可用参数，(而不是第0个)
    a = []
    for i in range(1, len(sys.argv)):
        a.append(str((sys.argv[i])))

    # 可用参数的第0个参数、第1个参数为 输入路径、与输出路径
    # 可用参数的第2个参数、第3个参数为 id、url，用于向后台发送程序运行状态
    input = a[0]
    output = a[1]
    id = a[2]
    url = a[3]

    # 复制下面一行在正式算法开始之前，用于向告知后台数据预处理程序开始运行
    requests.post(url, {'id': id, 'status': 1})

    # ----此处是数据处理的开始----
    func(input, output)
    # ----此处是数据处理的结束----

    # 复制下面一行在正式算法结束之后，用于向告知后台数据预处理程序运行结束
    requests.post(url, {'id': id, 'status': 0})

```

预处理结果保存过程，最好将列标题也一同保存(pandas中的 `header`)

数据预处理评估

数据预处理评估只支持6种操作： `>, =, <, >=, <=, 'in'` ，对同一列，执行可以执行多个操作，如此可以达到在一个区间范围内的要求。 `'in'` 指的是是不是在某离散集合。例如， `'John' in ['Annie', 'Bob', 'John']`

在预处理评估中，为防止列标题对评估结果产生影响，会将可能作为列标题的第一行删除，因此对于有列标题的数据结果没有影响，对于没有列标题的数据，第一行的数据将不再评估范围之内。