**数据科学基础大作业研究报告**

|  |  |  |  |
| --- | --- | --- | --- |
| 学号 | 姓名 | 邮箱 | Python练习完成题目数量 |
| 181250089 | 刘一铭 | 181250089@smail.nju.edu.cn | 199 |
| 181250129 | 檀潮 | 181250129@smail.nju.edu.cn | 200 |
| 181250146 | 王宇博 | 181250146@smail.nju.edu.cn | 200 |

**一、研究问题**

通过对学生在某类题目上的掌握程度及不同题目的相似性进行分析，针对每个同学在编程中的弱项，个性化推荐相应的编程练习题目。

**二、研究思路**

首先，寻找一个合适的度量指标，将学生对题目的掌握程度进行量化。我们认为，针对一道题目，所有同学的掌握程度应当服从正态分布，即掌握的很差或很好的学生人数较少，中间范围内的人数较多，所以还需要对找出的度量指标所对应的数据的正态性进行检验，当满足正态分布时，认为找出的该度量指标是合理的，可以用其作为衡量学生对题目掌握程度的标准。

**三、代码开源地址**

**四、研究方法**

1、生成源数据集

在对test\_data.json中原始数据的观察分析之后，我们决定按case\_id对test\_data.json中的数据进行分类，每个case\_id下都包含了所有学生在该题上的各自的提交数据，以此生成研究第一阶段所需的数据集。（进行该数据处理过程的Python文件为update\_data.py，生成的数据集为case\_data.json）

2、计算度量指标

数据约定：

“完成度”：得分与满分（100）的比值

“提交时间间隔”：两次提交时间差

指标一：program\_rate

计算方法：针对每一道题，筛选出提交次数四次及以上的学生的数据，对其中的每位学生，计算其在该题上的最大完成度和最大提交时间间隔的比值，然后对该值取自然对数，所得即为program\_rate值。

理由：我们认为最大完成度可以作为学生对题目的最终理解程度的一种度量，而提交时间的总跨度可以衡量学生的做题时间，包括think、program和debug的时间。

指标二：debug\_rate

计算方法：针对每一道题，筛选出提交次数三次及以上的学生的数据，对其中的每位学生，筛选出其提交记录中相邻两次提交时间间隔小于一小时且完成度有提升的记录，计算其中每两次提交完成度之差的平方的平均数，然后对该值开方并取自然对数，所得即为debug\_rate值。

理由：限定只计算完成度提升的记录数据，以此期望所求的数值有更多可能是在原方法下攻克用例期间得到，而不是在试错或者转换方法期间得到，使得行为与debug更相符；限定计算平均数，使得每次分数提升都作为测量的一部分，与program\_rate的测量标准相区别；限定只计算一小时内的分数提升，使得所测量的行为与debug更相符。

指标三：early\_success\_degree

计算方法：针对每一道题，筛选出提交次数四次及以上的学生的数据，对其中的每位学生，筛选出其提交记录中相邻两次提交完成度有提升的记录，计算记录中完成度的均值，即为early\_success\_degree值。

理由：对于一道题，分数提高的越早，可以体现编程者对题目的理解速度越快，故用每次满分程度的均值来刻画此指标，使得分数提高越早时高分的权重越大、此指标的值越大；而对于分数下降的时候，我们认为这是编程者在尝试别的方法，其对题目的理解程度应保持与原来不变，所以仍采用之前分数的最大值计算。

指标四：finish\_degree

计算方法：针对每一道题，筛选出提交次数四次及以上的学生的数据，对其中的每位学生，计算其每次提交时完成度（若本次提交完成度较之前下降，则按照之前提交记录中最高完成度计）与提交时间间隔（此处为首次提交时间与本次提交时间差）的比值的平均数，然后对该值取自然对数，所得即为finish\_degree值。

理由：在该指标下，越快得到越高的分数，那么该指标的值就越大，但是如果提交并没有带来分数的提升，即分数没有改变或甚至下降，那么就可以认为在该提上可能遇到了困难，同时由于时间增加而分数为提升，该指标的值也会降低，以此来描述学生对题目的完成速度，间接反映其对题目的掌握程度。

3、生成中间数据集

依次计算每道题下每位学生做题数据的上述四个指标的具体数值，去除无效数据，并筛选出符合“含有有效指标个数为15及以上”条件的case\_id，然后将所得数据以四个指标进行分类，写入intermediate\_case\_data.json文件中，作为研究第一阶段的中间数据集。

4、正态性检验

我们采用偏度、峰度检验的方法来检验中间数据集中数据的正态性。（偏度、峰度检验代码见normalDistributionTest.py文件）

分别针对上述四个指标，进行如下验证：

计算步骤：

Ⅰ.取显著性水平α为0.004，则zα/4 = z0.001 = 3.09，n为相应指标下的数据数量

Ⅱ.使用如下公式计算σ1，σ2，μ2

σ1 = ，σ2 = ，μ2 = 3 -

Ⅲ.计算样本中心距B2，B3，B4，利用以下关系式

B2 = A2–A12，B3 = A3–3A2A1 + 2A13，B4 = A4–4A3A1 + 6A2A12–3A14

其中，Ak = (k = 1,2,3,4)为k阶原点矩

Ⅳ.计算偏度、峰度的观察值g1，g2

g1 = ，g2 =

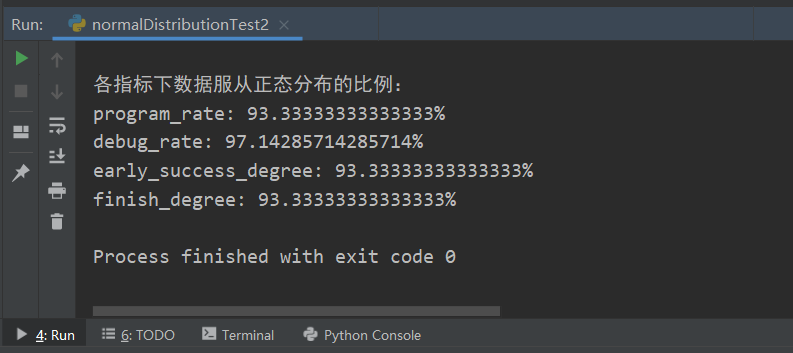
Ⅴ.确定拒绝域

由于zα/4 = z0.001 = 3.09，则拒绝域为|u1|=|g1/σ1|≥3.09或|u2|=|g1-μ2|/σ2≥3.09

Ⅵ.判定样本数据是否符合正态

假设算得|u1|＜3.09，|u2|＜3.09，那么认为数据服从正态分布，否则认为不服从

验证结果：



经偏度、峰度检验，上述四个度量指标下的数据服从正态分布的比例均在90%以上，由此我们认为所选的度量指标较为合理，可以作为刻画学生对于题目掌握程度的度量标准。