数据分析与实践实验 3

LaTeX by 常征 PB23030850

2025 年 4 月 24 日

目录

1 （30%）使用 pandas 库的相关方法，进行数据集读取、信息处理和探索性分析。

感觉这个实验就一直在用库里的各种方法，不需要自己写算法。

1.1 读取数据集 data.csv（将首列作为索引）至变量 df，展示该数据集的前 10 行内容，并展示数据集有多少行和多少列。

pandas 提供了 read_csv 方法，我们直接调用即可读取数据集。输出前 10 行，可直接用 head 方法，输出列可用 shape 方法。

```
前十行内容------------------------------
    CNTRYID  CNT  CNTSCHID   CYC  NatCen  Region  STRATUM  SUBNATIO  OECD  \
1         8  ALB    800002  07MS     800     800  ALB0109     80000     0
2         8  ALB    800003  07MS     800     800  ALB0109     80000     0
3         8  ALB    800004  07MS     800     800  ALB0211     80000     0
4         8  ALB    800005  07MS     800     800  ALB0107     80000     0
5         8  ALB    800006  07MS     800     800  ALB0105     80000     0
6         8  ALB    800007  07MS     800     800  ALB0109     80000     0
7         8  ALB    800008  07MS     800     800  ALB0210     80000     0
8         8  ALB    800009  07MS     800     800  ALB0203     80000     0
9         8  ALB    800010  07MS     800     800  ALB0210     80000     0
10        8  ALB    800011  07MS     800     800  ALB0206     80000     0

    ADMINMODE  ...  EDUSHORT  STAFFSHORT  STUBEHA  TEACHBEHA  SCMCEG  \
1           2  ...    1.2478     -1.4551  -1.1797    -2.0409 -1.0391
2           2  ...    0.6221     -1.4551   2.1196     2.5150 -1.0542
3           2  ...    0.4591     -1.4551  -0.6199    -0.4415  0.9042
4           2  ...    1.3065      1.2432  -0.2682    -0.0452 -1.5648
5           2  ...   -0.2376     -1.4551  -1.3196    -2.0409  0.9042
6           2  ...    0.8339      0.0055  -3.3785    -0.9173  0.1329
7           2  ...   -1.4212     -1.4551  -2.0719    -2.0409  0.9042
8           2  ...    0.1000     -0.0963   0.6482     1.3992  0.1329
9           2  ...   -1.4212     -1.4551  -2.0719    -2.0409  0.9042
10          2  ...   -1.4212     -0.5869  -0.9148    -0.6289 -2.5190
...

[10 rows x 197 columns]
行数/列数------------------------------
21903 197
```

图 1: 1.1 结果

1.2 数据集存在很多缺失值，输出各列缺失值的个数，并删除数据集的最后一列。基于更新后的数据，展示哪一列的缺失值最多，哪些列没有缺失值。

判断缺失值我们可以用 isnull 方法，由于我们是一列一列的考察，因此可以用 sum 方法把 dataframe 类型 (就是一个二维表) 转化成 series 类型 (就是一个数组)，删除的话就直接用 drop 方法就行了。

比较好玩的是 Pandas 库里提供了布尔索引，比如

```
miss_value[miss_value == max_missing]
```

就会返回所有满足条件 miss_value == max_missing 的元素。

最后用 tolist，可以把 series 类型转化为 list 类型.

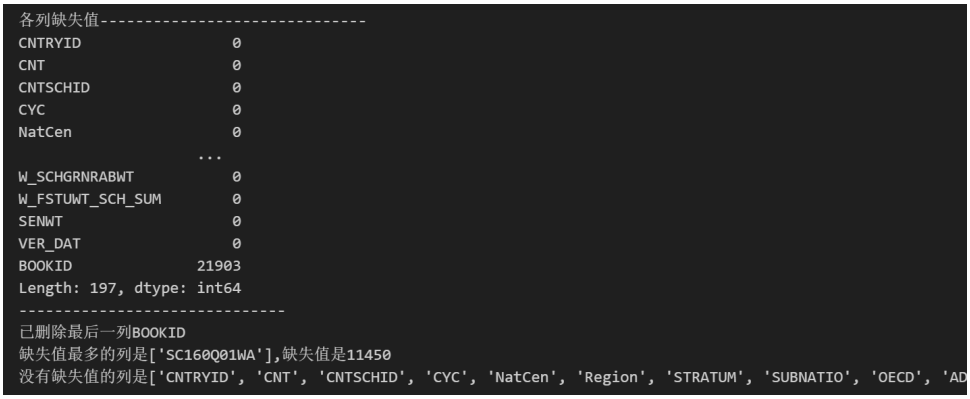

图 2: 1.2 结果

1.3 数据集的有些列在所有记录上均有相同取值，作为独立的一列是相对冗余的。请查找并输出这些列的名称和取值，并阐述这些列代表的含义，最后删除这些列。

对于是否有相同的取值，我们可以用 nunique 方法来统计不同取值的数量，如果某一列的结果是 1，那么就说明只有 1 个值。

图 3: 1.3 结果

cyc 代表 cycle，其取值是 07ms 说明所有人都以 0.7ms 为一个循环。

ADMINMODE 代表管理员模式，其取值都是 2 可能说明大家都是同一个身份 (普通用户/管理员)。

1.4 观察 PRIVATESCH 特征列，统计所有取值及其出现的次数。其中有一部分取值含义一致但形式不同（如 private 和 PRIVATE），试对它们进行归并，随后展示所有取值及其出现的次数。

首先我们先把所有取值和次数都输出一遍，这需要用到 value_count 方法。

```
原始PRIVATESCH列的所有取值及出现次数------------------------------
PRIVATESCH
public     12161
missing     5295
private     3443
invalid      251
PRIVATE       84
PUBLIC        73
Name: count, dtype: int64
```

图 4: 原始取值

我们发现主要影响就是大小写问题，所以归并的时候统一把大写转化为小写就行了 (invalid 和 Missing 应该不用归并成一类吧)。我们先用 str，表示对其中字符串进行操作；再使用 lower 方法把大写降为小写。

```
原始PRIVATESCH列的所有取值及出现次数------------------------------
PRIVATESCH
public     12161
missing     5295
private     3443
invalid      251
PRIVATE       84
PUBLIC        73
Name: count, dtype: int64
归并后PRIVATESCH列的所有取值及出现次数------------------------------
PRIVATESCH
public     12234
missing     5295
private     3527
invalid      251
Name: count, dtype: int64
```

图 5: 1.4 结果

1.5 选取特征 STUBEHA,TEACHBEHA,EDUSHORT,STAFFSHORT，展示它们的基本统计特征信息 (平均值、标准差、四分位点、最小值、最大值、Pearson 相关系数矩阵)。

首先我们需要从原始 dataframe 里选出这几个特征，选出之后直接用 discribe 方法就可以啦。discribe 方法包括平均值，标准差等等，只少了题目要求的 Pearcson 相关系数矩阵。

想计算相关系数矩阵，我们要使用 corr 方法，并且指定参数 method = 'pearson'

	STUBEHA	TEACHBEHA	EDUSHORT	STAFFSHORT
count	20863.000000	20846.000000	20752.000000	20765.000000
mean	0.041614	0.108233	0.120716	-0.013901
std	1.236531	1.158154	1.091434	1.059587
min	-4.354200	-3.239200	-1.931900	-2.589100
25%	-0.682300	-0.621800	-0.688400	-0.782800
50%	0.041700	0.226600	0.100000	0.013100
75%	0.815300	0.852425	0.833900	0.673600
max	3.627400	3.833800	3.522900	4.112500

	STUBEHA	TEACHBEHA	EDUSHORT	STAFFSHORT
STUBEHA	1.000000	0.633862	0.239674	0.257259
TEACHBEHA	0.633862	1.000000	0.215399	0.331982
EDUSHORT	0.239674	0.215399	1.000000	0.483617
STAFFSHORT	0.257259	0.331982	0.483617	1.000000

图 6: 1.5 结果

1.6 Q5 中所得 Pearson 相关系数矩阵显示，特征 STUBEHA 与 TEACHBEHA 之间、EDUSHORT 与 STAFFSHORT 的相关系数较高，请通过特征定义推测可能导致相关性的原因。

可能是因为这些特征具有一定的因果关系。

STUBEHA 指学生行为，TEACHBEHA 指教师行为。如果学生的行为好，可能对教师行为具有激励效果；反之，教师的行为也会影响学生行为。

EDUSHORT 指教育短缺，STAFFSHORT 指人员短缺。教育资源短缺说明可能没有足够资金，而不充足的资金也无法招聘到足够的教职工。二者虽然不具有因果，但是可能是被同一原因导致。

1.7 以特征 PRIVATESCH 为先验条件，对其余各特征中可能存在的缺失值进行均值填补。

前面我们已经处理过 PRIVATESCH，它只有几种有限的离散取值情况，所以我们采取**分组均值**填补。

既然是分组均值，首先我们就得分组，而 pandas 提供了 groupby 方法，用于分组。

在填补之前，我先执行了以下命令，用于输出我们即将填补的均值

```
group_mean = df1.groupby('PRIVATESCH').mean()
print('各组均值'+'-'*30)
print(group_mean)
```

对于正式的填补，我使用了 transform 函数，对每一个分组应用一个函数，而函数使用

```
lambda x:x.fillna(mean(x))
```

填入，即对所有 NAN，填补 mean(x)

```
各组均值------------------------------
             EDUSHORT  STAFFSHORT
PRIVATESCH
invalid     -0.055418   -0.349525
missing      0.269301   -0.188506
private     -0.425706   -0.314380
public       0.229345    0.144431
------------------------------
已填补
```

图 7: 1.7 结果

2 导入 numpy 和 matplotlib 库，对数据集 df 进行一定数据可视化分析。

2.1 选择两个连续数值型特征，绘制其分布散点图，要求合理设置散点颜色和大小，并配上合适的标题和图例注记。

这个需要用到 matplotlib 库。为了方便起见，我们不妨就选取上面已经知道有一定关系的'TEACHBEHA'和'STUBEHA'，分别让它们做 y 轴和 x 轴。不过在开始前，我们需要先去除 nan，那么就要用到 pandas 的 dropna 方法。

剩下的就是设置图表的各种参数，比如 subplots 设置表的大小，scatter 设置散点的各类参数，set_title 设置表的标题，set_xlabel 设置 x 轴标题，legend 设置图例。

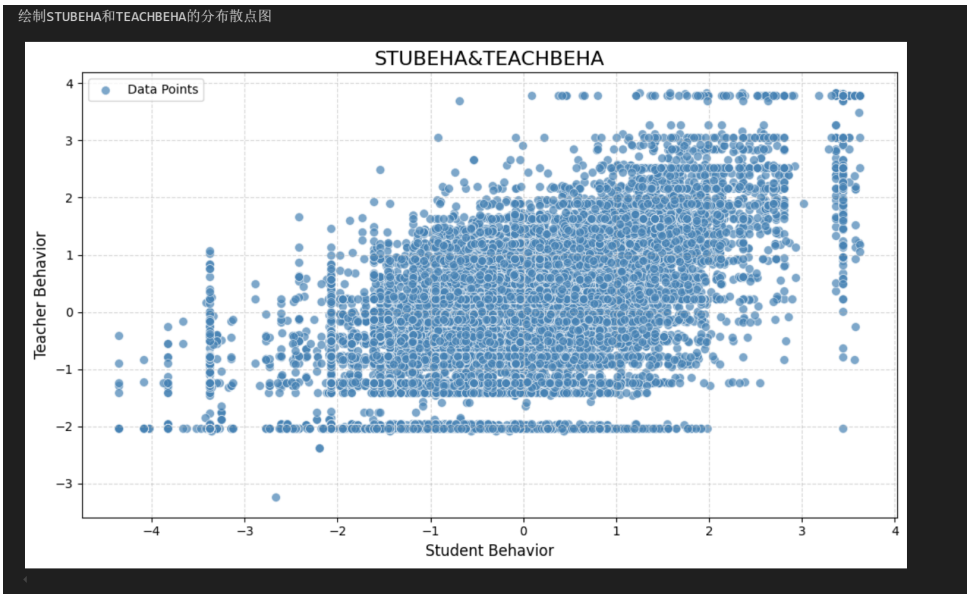

图 8: 2.1 结果

2.2 选择一个离散数值型特征（建议所有取值数量不超过 10），绘制饼图，要求设置合理配色和比例，并配上合适的标题和图例注记。

我选择”SCHLTYPE”。不过同样，我们需要先用之前用过的 value_count 来统计各类 value 的值，再用 Lable 和 value 分别存储 lable 和数量。

剩下的和上面大致相同，设置图表的各种参数。不过有一点不同。

我们需要为饼状图的每一块设置一个合适的颜色，这就需要一个映射，使用到了 matplotlib.cm，并且用了 numpy 进行切片。

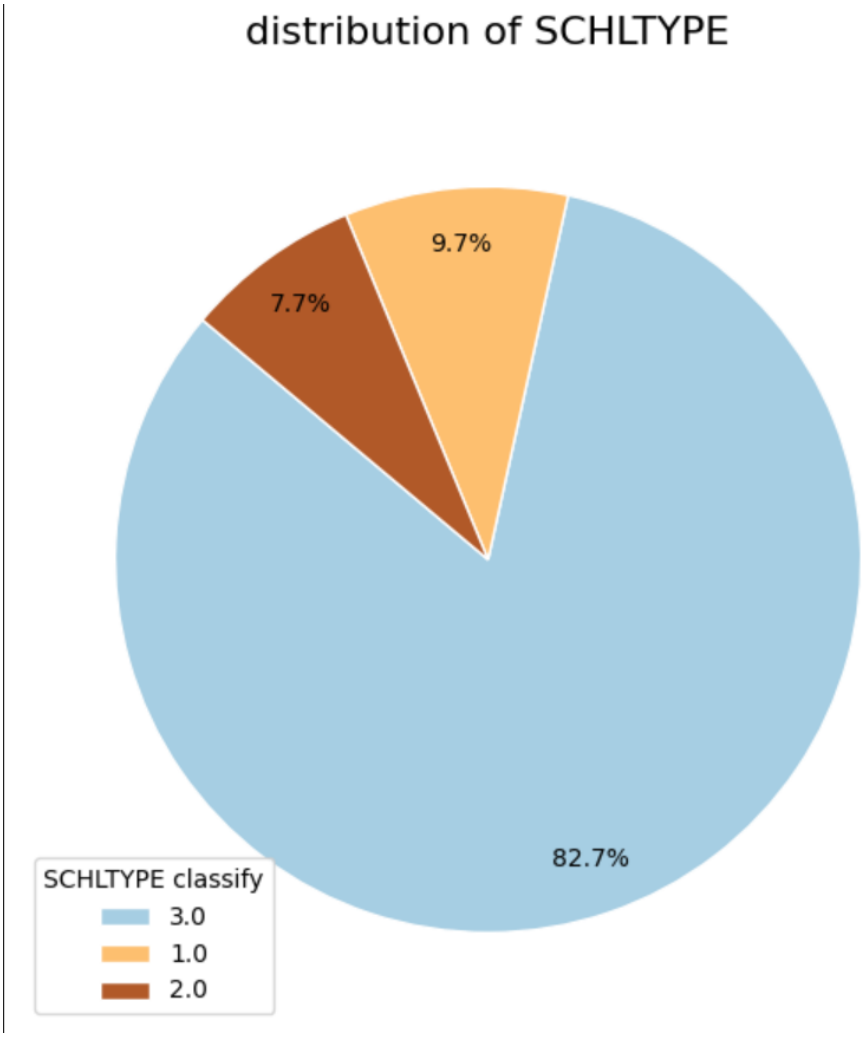

图 9: 2.2 结果

2.3 对 T1-Q5 中的 Pearson 相关系数矩阵，绘制热力矩阵图，要求为每个位置增添对应数值表示（保留三位小数），设置数值与颜色的对应关系条，并配上合适的标题和坐标表示。

这题让我惊讶几个库的协作性。

最关键的函数是 matplotlib 的 imshow，它将 numpy 数组显示为彩色。而 numpy 数组又是我们直接对上面用 pandas 库里的 value 方法得到的。

设置颜色条和标签，我们直接使用 colorbar 方法得到 color_bar 对象，对这个对象用 set_label

方法。

设置坐标轴得用 set_xticks 和 set_xtickslabels，和上面不一样是因为这一次是矩阵，而非一个轴。

最后，为了在每一个上标注数据，我们需要一个双层的嵌套循环，通过循环遍历每一个格子，逐个标注数据。

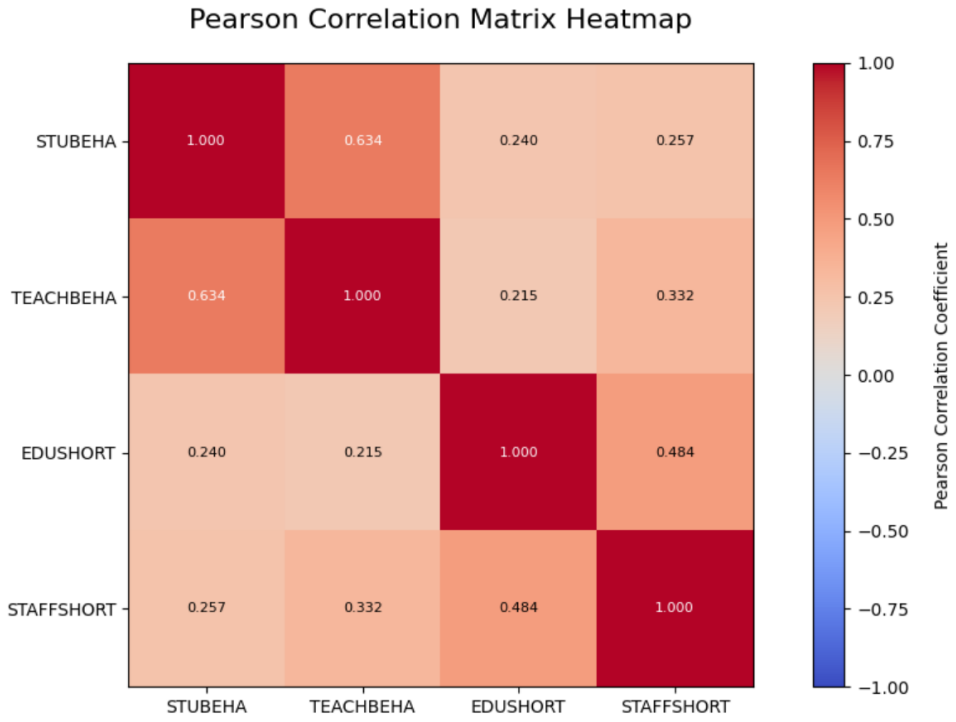

图 10: 2.3 结果

3 现欲对数据集特征 STUBEHA,TEACHBEHA 进行分布校验。执行以下子表提取和缺失记录删除操作，并基于 df2 完成以下任务：

3.1 以区间数为 10，分别绘制两个特征的频数直方图，基于频数直方图的结果，是否可以认为两特征近似服从正态分布？

这一题需要我们对 subplots 方法设置不同的参数,以前我们的调用都是 plt.subplots(figsize=(10, 6))，只设置了 figsize 参数。但是现在由于我们需要同时绘制两个图，因此需要把参数设置为 plt.subplots(1, 2, figsize=(10, 6)) 表示一行两列，这样绘制出的图就是并排的。

此外，和上面不同的还有一点，之前都是直接对 ax 调用方法，但是这一次会返回两列，所有有两个 ax，因此要对 ax[0], ax[1] 分别调用方法设置参数

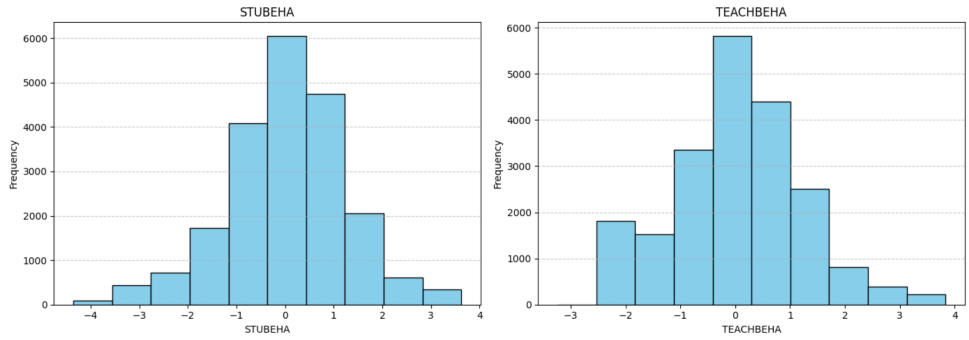

图 11: 3.1 结果

我认为 STUBEHA 近似服从正态分布，TEACHBEHA 不服从。

3.2 分别绘制这两个特征样本的 Q-Q 图，基于 Q-Q 图的结果，是否可以认为两特征近似服从正态分布？

如果我们要画出 QQ 图来验证是否服从正态分布，那么我们首先得得到正态分布，而得到正态分布的方法就是用 scipy 库，并且规定 dist="norm"。

真的好神奇，加入了 scipy 库之后几个库的函数都可以相互传递参数，太伟大了 python。

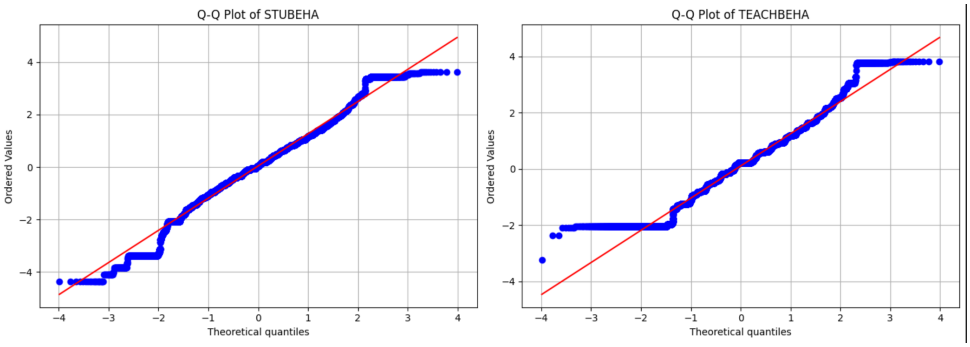

图 12: 3.2 结果

基于 Q-Q 图的结果，可以认为两特征都近似服从正态分布。

对比上一问，感觉肉眼判断还是不够准。比如我看 TEACHBEHA 的最左边高了很多，但是在 qq 图上也没有高那么多。

3.3 请自行编写代码绘制两特征样本的 Q-Q 图和直线 y=x，并基于可视化结果简述你的发现。

首先和上面类似，我们使用 value 方法得到一个 Np 数组，而以后的绘图都是对 np 数组进行操作。

为了尽可能准确的比较，我们应该比较对应百分位的值，而非对应位置的值。比如我们应该比较 TEACHBEHA 和 STUBEHA 的第 10% 百分位的值，而不是比较他们第 10 个的值。

我们使用 np.linspace(1, 99, 100) 得到 1% 到 99% 的 99 个点，再用 percentile 方法得到样本各个百分位的值。

接下来画 qq 图，和上面类似，不过额外的，我们还需要画一条 y=x 线。它的起点可以是 qq 图最左下角的值，终点是 qq 图最右上角的值，剩下的就是画一条直线。

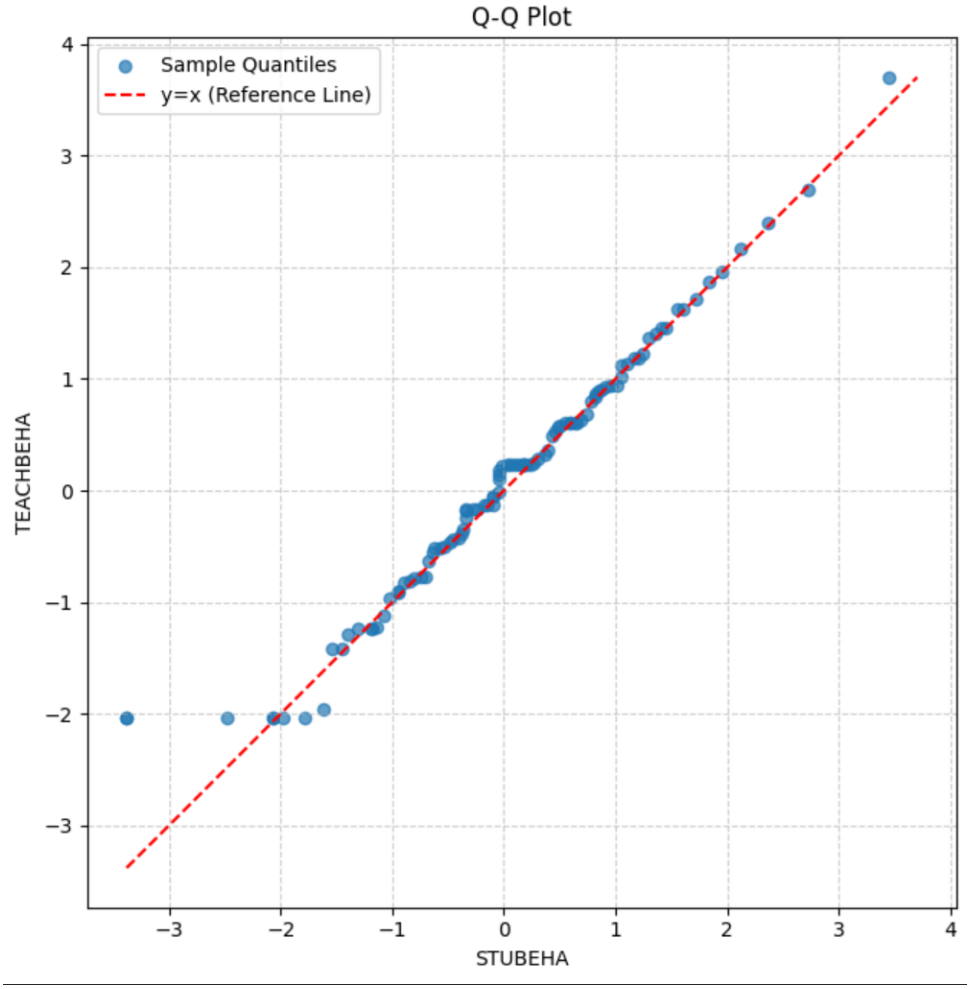

图 13: 3.3 结果

我发现虽然他们的相关系数只有 0.6，看上去没有特别高，但是在 qq 图上却非常贴近 y=x，从这个角度来看很它们的理论分布具有很强的一致性。

4 基于正态分布假设，对特征 STUBEHA,TEACHBEHA 的总体分布进行参数估计。

4.1 请分别求均值参数和方差参数的极大似然估计。

下面我们只推到 STBEHA，因为 TEACHBAHA 的推导式一样的。因为服从 $N(\mu_1, \sigma_1^2)$，所以概率密度函数为 $f(x) = \frac{1}{\sqrt{2\pi}\sigma_1} e^{-\frac{(x-\mu_1)^2}{2\sigma_1^2}}$。

那么计算极大似然估计时，就有

$$L(X; \mu_1.\sigma_1) = \Pi f(x_i) \tag{1}$$

取对数，就有

$$l(X;\mu_1.\sigma_1) = \sum \ln\ f(x_i) = \sum[-\ln(\sigma_1) - \frac{(x_i-\mu_1)^2}{2\sigma_1^2}] = -n\ln(\sigma_1) - \frac{\sum(x_i-\mu_1)^2}{2\sigma_1^2} + C \quad (2)$$

先对 μ_1 求偏导，令偏导等于 0，就有：

$$\frac{\partial \ln f}{\partial \mu_1} = \frac{\sum(\mu_1 - x_i)}{\sigma_1^2} = 0 \quad (3)$$

解得

$$\mu_1 = \frac{1}{n}\sum x_i \quad (4)$$

在对 σ_1 求偏导，令偏导等于 0，就有：

$$\frac{\partial \ln f}{\partial \sigma_1} = -n\frac{1}{\sigma_1} + \frac{\sum(\mu_1 - x_i)^2}{\sigma_1^3} = 0 \quad (5)$$

解得

$$\sigma_1^2 = \frac{1}{n}\sum(x_i-\mu_1)^2 = \frac{1}{n}\sum(x_i-\bar{x})^2 \quad (6)$$

所以，μ_1 的极大似然估计是 $\bar{x}$，其中 x 是 STUBEHA 的取值；μ_2 的极大似然估计是 $\bar{y}$，其中 y 是 TEACHBEHA 的取值。

σ_1 的极大似然估计是 $\frac{1}{n}\sum(x_i-\bar{x})^2$，其中 x 是 STUBEHA 的取值；$\sigma_2$ 的极大似然估计是 $\frac{1}{n}\sum(y_i-\bar{y})^2$，其中 y 是 TEACHBEHA 的取值。

具体的代码实现就用 Numpy 的对应方法，比如用 mean 求均值，用 sum 求和。

```
STUBEHA均值:0.04161388103340842, 方差:1.5289359134956648
TEACHBEHA均值:0.10823287441235728, 方差:1.341256978690452
```

图 14: 4.1 结果

μ_i 是无偏估计，因为 $E\bar{x} = \mu_i$，满足无偏性要求。

σ_i 不是无偏估计，因为 $E\frac{1}{n}\sum(x_i-\bar{x})^2 = \frac{n-1}{n}\sigma$，不满足无偏性要求。

4.2 对该特征进行常数估计，求参数的最小二乘解，并比较其与 Q1 中所得总体均值极大似然估计的结果。

求常数估计的最小二乘，就是求

$$a = argmin\{\sum(y_i - a)^2\} \quad (7)$$

我们对 a 求偏导，就有 $\sum 2(a-y_i) = 0$，同样解得 $a = \bar{y}$。和上面结果一样！

5 基于 T4 的假设，现需对特征 STUBEHA,TEACHBEHA 的总体均值差异进行检验。请阅读本文档的内容，并导入 scipy 库，完成以下任务：

5.1 简述本情景下应使用成组检验还是成对检验，并写出单侧检验原假设。

这两个特征具有一定的相关性，所以我认为应该采取成对检验。

单侧检验有两种。检验学生行为均值是否大于教师行为均值时，原假设可以写作：

$$H_0 : \bar{S} - \bar{T} \leq 0 \tag{8}$$

对立假设是：

$$H_1 : \bar{S} - \bar{T} > 0 \tag{9}$$

检验学生行为均值是否小于教师行为均值时，原假设可以写作：

$$H_0 : \bar{S} - \bar{T} \geq 0 \tag{10}$$

对立假设是：

$$H_1 : \bar{S} - \bar{T} < 0 \tag{11}$$

5.2 使用 scipy.stats 中的相关方法，执行相应的假设检验。

根据我们之前得到的数据，我们发现 TEACHBEHA 的均值更大，所以我们给出的原假设是 TEACHBEHA 的均值减 STUBEHA 的均值大于 0。

我发现不可以用 dropna 方法来去除 nan，因为这样就会导致列数不对应。需要在 ttest_rel 方法里指定 nan_policy='omit'

```
对原假设H0为STUBEHA>TEACHBEHA进行成对假设------------------------------
P-value: 2.4682470543872093e-21
```

图 15: 5.2 结果

5.3 基于 Q2 所得结果，请仔细斟酌并叙述你所得到的结论。

检验结果得到的 P 值为 2.468e-21(看见这么小不太放心，但是 ai 说在大样本下这么小是可能的)。由于该 P 值小于显著性水平 $\alpha = 0.05$，我们可以拒绝原假设 $H : {}_{S}TUBEHA \geq {}_{T}EACHBEHA$。因此，我们有足够的统计证据表明，在该数据集所代表的总体中，TEACHBEHA 的平均水平显著高于 STUBEHA 的平均水平。

5.4 上述结论隐含了犯哪一类错误的可能？相应犯错概率是多少？

隐含犯第一类错误的可能，犯错概率不超过 $\alpha = 0.05$