Question quality assessment

The task is to estimate the quality of questions from student responses and other information about questions, using the training data provided in the "train_data" subfolder. Then we must calculate the rankings of the questions using the template in "submission/template.csv". The validation data is provided in the "test_data" subfolder, where the "quality_response_remapped_public.csv" file should be used for validation, and the "quality_response_remapped_private.csv" file should be used for testing.

The overall objective is to use machine learning techniques to accurately estimate the quality of questions in educational assessments, based on student responses and other relevant information.

The problem here is to assess the quality of questions to ensure that assessments are fair and reliable.

Thanks to the “task3_eda.ipynb” we know each question’s mean of correct answers, its standard error of mean and we also know how 5 students judged the quality of the question. Now that we have a dependent variable and an independent variable, we will be able to use a linear regression model to see and use it to predict the test data and see if the success rate impacts the mean of the quality of quality_response files.
Finaly we will calculate the score all while considering the potential bias.

- Introduction :

The goal of this report is to estimate the quality of questions from student's responses and other information about this set of questions.

From this list of question we will calculate a score that will allow us to rank the 948 questions (see the 20228354.csv file)

For that, we will need to figure out how to judge the quality of questions, what are the parameters defining it and what informations do we already have.

- Methods:

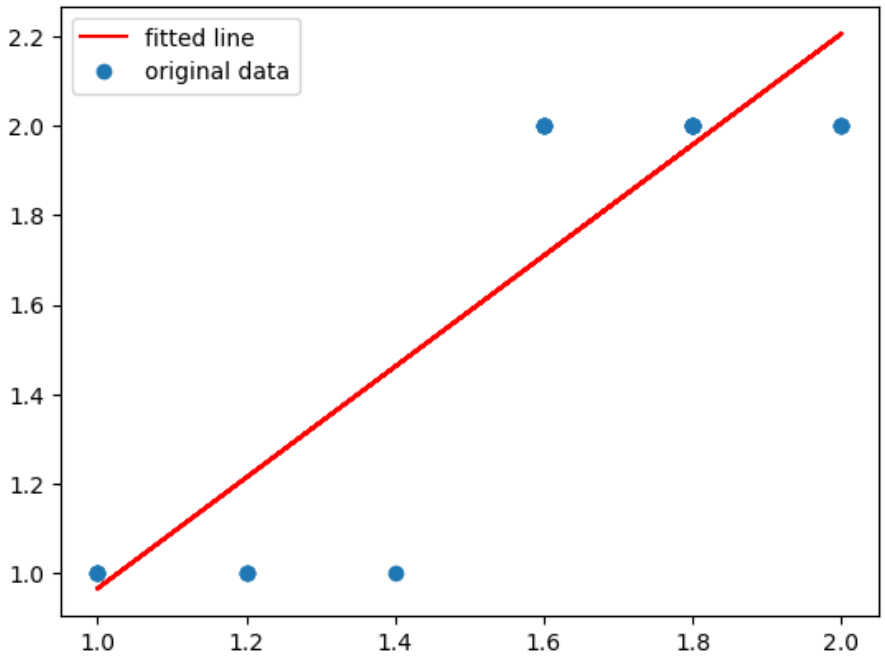
linear regression lines allow us to distinguish pattern among the student’s judgment, they tend to give better grades to the questions they succeeded

```
x = eval_validation['score']
y = eval_validation['preference']
# linear regression
slope, intercept, r_value, p_value, std_err = linregress(x, y)

print(f"Slope: {slope}")
print(f"Intercept: {intercept}")
print(f"R value: {r_value}")
print(f"P value: {p_value}")
print(f"Standard error: {std_err}")
plt.plot(x, intercept + slope*x, 'r', label='fitted line')
plt.plot(x, y, 'o', label='original data')
plt.legend()
```

```
Slope: 1.2417400881057272
Intercept: -0.27698237885462595
R value: 0.9075052906470987
P value: 3.8954428153045336e-10
Standard error: 0.11984197015171474
```

[53]: `<matplotlib.legend.Legend at 0x16cb8108>`



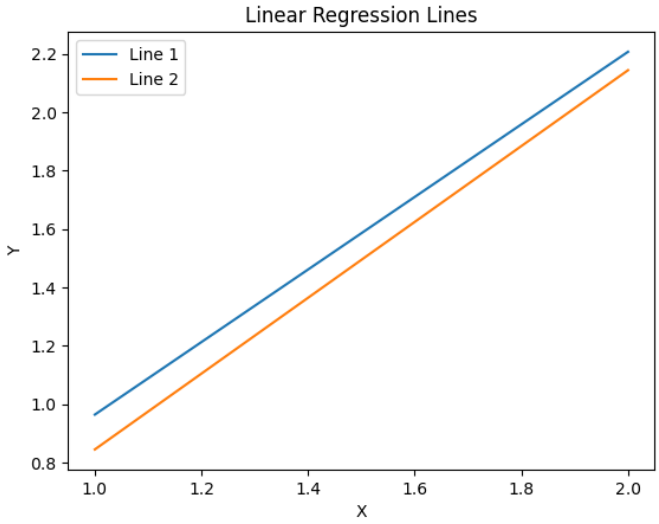
P value is way below 0.05 so there is a strong relationship

```
[113]: import matplotlib.pyplot as plt

x_range = [1, 2]
slope1 = 1.2417400881057272
intercept1 = -0.27698237885462595
y1 = [slope1 * x + intercept1 for x in x_range]
slope2 = 1.2979574937896763
intercept2 = -0.45242202594534775
y2 = [slope2 * x + intercept2 for x in x_range]
plt.plot(x_range, y1, label="Line 1")
plt.plot(x_range, y2, label="Line 2")

plt.title("Linear Regression Lines")
plt.xlabel("X")
plt.ylabel("Y")
plt.legend()
plt.show()

template = pd.read_csv('submission/template.csv')
len(template)
```



The results obtained with the test and the validation seems relatively similar and have a strong relationship.

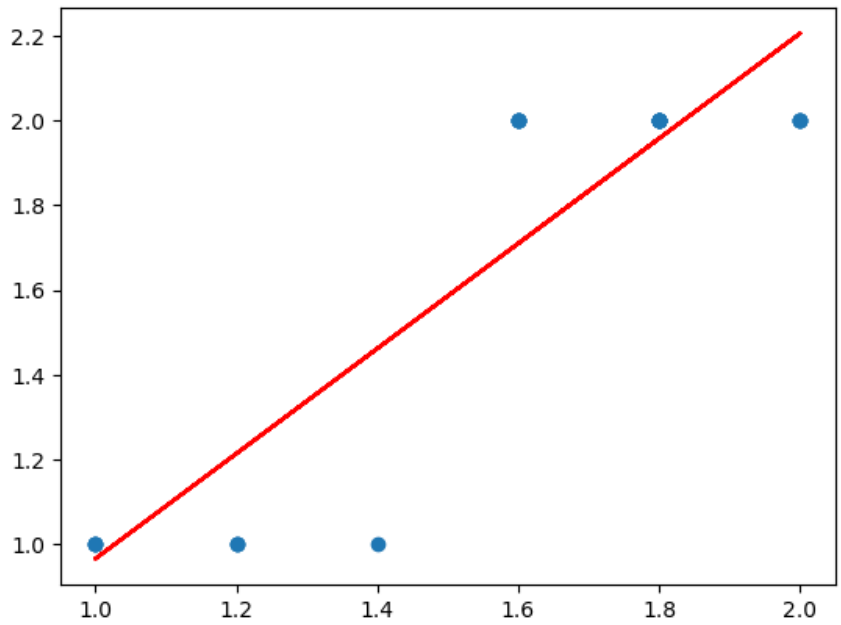
We can now do the same with the mean of correct answers, its standard error of mean. To do that I calculated a score considering the standard error of mean and merged eval_validation with data to then do a linear regression.

Repository:
https://github.com/N3ROD/ML_Soongsil_university.git

- Results:

After calculations we get the following:

```
Slope: 1.2417400881057272
Intercept: -0.27698237885462595
R value: 0.9075052906470987
P value: 3.8954428153045336e-10
Standard error: 0.11984197015171474
```



Slope: 1.2417400881057272
Intercept: -0.27698237885462595
R value: 0.9075052906470987
P value: 3.8954428153045336e-10
Standard error: 0.11984197015171474

R^2 is between 0 and 1 and is very close to 1 which is very good in terms of explanations for the prediction.
P value is extremely low as well so the results are replicable.
Finaly by using the average of correct answers and the standard error of mean and then ranks the questions from best to worst score and saves them in the 20228354.csv file with the QuestionId and ranking columns:

- Conclusion :

Thus, by using simple linear regression, least squares fit, standard error and P value as well as R^2 and some research (see the references) I was able to estimate the quality of questions from student's responses and other information about questions quality related to their rate of success with the goal of making my prediction as much reliable as possible to at the end calculate the rankings of the questions accurately.

- Future directions:

The quality of test questions or items can be measured with the help of "Item Analysis Process". For item analysis, we must calculate the reliability and validity of the test. For each item or question, two important measures are calculated. These are:

1. Difficulty value of each item or question

2. Discriminative Value of each item or question.

After some research I found out that the best difficulty for a Four-response multiple-choice is 74% of success. We can get it by adding up the number of points earned by all students on the question and dividing that total by the number of students.

As for the Discriminative value, it is obtained by subtracting the number of students in the lower group who got the question correct from the number of students in the upper group who got the question correct. Then, divide by the number of students in each group.

Once we have calculated these values for each question, we will be able to add those new variables in our predictions and/or compare its results with the ranking obtained.

- REFERENCES:

https://www.researchgate.net/post/How_can_the_quality_of_test_questions_or_items_be_measured#:~:text=Probably%20you%20can%20determine%20quality,%2C%20reliability%2C%20objectivity%20and%20clarity

https://www.washington.edu/assessment/scanning-scoring/scoring/reports/item-analysis/

https://fcit.usf.edu/assessment/selected/responsec.html#:~:text=Determine%20the%20Discrimination%20Index%20by,are%20five%20in%20each%20group)

https://jakevdp.github.io/PythonDataScienceHandbook/03.07-merge-and-join.html

https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.linregress.html

https://hastie.su.domains/ISLR2/ISLRv2_website.pdf