Before we start working with this toy dataset, I really want to say that I know this dataset before, we can find this dataset from Kaggle or UCI Machine Learning Repository. There should be another csv file called ‘’trial’’ in the original dataset with 18 columns.

**Exploration of Data Analysis**

1. The variable title “Score\_B” appears 2 times, but their data is not the same

2. The variable “Detection\_Risk” only has one kind of variable, so we do not need to take it into account.

3. There are two kind of labels in this dataset. One is “Risk” which is classification label, the other one is “Audit\_Risk” which is regression label. We can do these two tasks based on these two labels.

A screenshot of a cell phone

Description automatically generated

4. Threshold to trigger risk: I believe there should be a threshold to classify whether an Audit\_Risk is Risk. So I write a program to find the threshold number and find if Audit\_Risk is larger than 1.0144, it will be classified as 1; otherwise it will be classified as 0. To prove this, I write another program to see whether there is an outlier and find there is no outlier.

A screenshot of a cell phone

Description automatically generated

5. Weight of each variable: When we get a new dataset with many columns of variables, one thing we want to know most is which column is the most important one. For the problem of classification, we can use KL divergence or logistic regression to calculate the weight of each variable. For the problem of regression, we can use Pearson correlation coefficient to get the weight of each variable. So I calculate the weight with the later one:

A close up of text on a black background

Description automatically generated

We can see that the weights of “PARA\_B”, “Risk\_B”, “TOTAL” and “Inherent\_Risk” are very high compared with the others, so these variables are the most important variables.

**Data preprocessing**

We use decision tree model in sklearn to classify our data. To make it, we have to make sure all of our input in number instead of string. I write a “is\_number” function to distinguish whether a string can transform into number. I find that 3 strings are made of English letters and the other 1 is space. So the count of no number is only 4, I use 0 to replace them as follows:

A screenshot of a cell phone

Description automatically generated

**Modelling and predict with Decision Tree**

The number of the dataset is 776, I use 676 of them as training dataset and 100 of then as testing dataset. I use the Decision Tree model in sklearn as my classification model:

A screenshot of a cell phone

Description automatically generated

At first, you may think this result means overfitting. But considering that our dataset is very small, so I think this result is acceptable.