REPORT : Book Rating Prediction Model Python Project

Instructor : **Hanna Abi Akl**
Project made by : **Ali Semlali**

For this project, I used Jupyter notebook with Python coding. I also did the project hosting on Github with a **README file**.

Github link for the project : https://github.com/SemlaliAli/Books-Prediction-Python-Project

In this project, we are going to analyze a database of books, in order to build a model that predicts the rating of books.

To carry out this project, we will proceed as follows:

- First, we will explore our database: find out how **many columns** it contains, find out the **type of information** that exists, find out the **data types** of each column, find out the **maximum**, **minimum and average values** of the columns that contain numerical data types.

- Then we will proceed to check the **integrity of the data** and see if there are null values in each column and proceed to **clean the data** and plot some relevant attributes.

  To clean the data, we corrected the values of the **language_code**, and the name of the column **num_pages.**

  Outliers are also problematic for our data set. They can distort statistical analyses and violate their assumptions, so we had to remove them. After we did some useful attributes plot like : **Histograms concerning average_rating and num_pages, number of books per rating …etc**

- Second, we did some features selection to aid us in our mission to create an accurate predictive model. Then, feature pruning with encoding some Data like book title and authors.

- To make the predictions of the books' average ratings, we used 4 models :

  1. **LinearRegression()**
  2. **KNeighborsRegressor()**
  3. **DecisionTreeRegressor()**
  4. **GradientBoostingRegressor()**

  For model evaluation, we used some metrics like : **mean_absolute_error** and **mean_squared_error.** We also used the **model score**.

  From the results we got, we can say that the linear regression model is not very accurate, but the predicted values and the actual values are close to each other. We can also see that DecisionTreeRegressor had the best model score with good accuracy between the actual and predicted values. For the GradientBoostingRegressor, we had the least values of EAM = 0,22 and ECM = 0,11. On the other hand the model training score is about 35 % wich is less than the DecisionTreeRegressor. Otherwise, the results we had in terms of predictions are satisfactory.