Machine learning python Lab

Data scientech institute 2021/2022

Auteur :
Warren LATA

Professeur :
Hanna Abi Akl

26 June 2022

1 Introduction

This report is an overview of my work on python Project. I will detail the differents step of my analysis, the result and the dockerization of the project on the main lines.

2 The data

The dataset is about published books and contains informations about some books. These informations are their relative features.

As attributes we have :

— A unique identification number for each book. BookId
— title The name under which the book was published.
— authors : The names of the authors of the book. Multiple authors are delimited by “/”.
— average_rating : The average rating of the book received in total.
— isbn : Another unique number to identify the book, known as the International Standard Book Number.
— isbn13 : A 13-digit ISBN to identify the book, instead of the standard 11-digit ISBN.
— language_code : Indicates the primary language of the book. For instance, “eng” is standard for English.
— num_pages : The number of pages the book contains.
— ratings_count : The total number of ratings the book received.
— text_reviews_count : The total number of written text reviews the book received.
— publication_date : The date the book was published.
— publisher : The name of the book publisher.

The project goal was to train a model to predict the books rating. I will now proceed to the different steps of my modelisation.

3 Data Processing

The dataset has initially 11123 observations and no missing values.Afte some explorations i have discovered that the average_rating was normally distributed and the numerical features were skewed. As espected the idenfiers as BookId, isbn was unique and nearly 80% of the books was written in english.

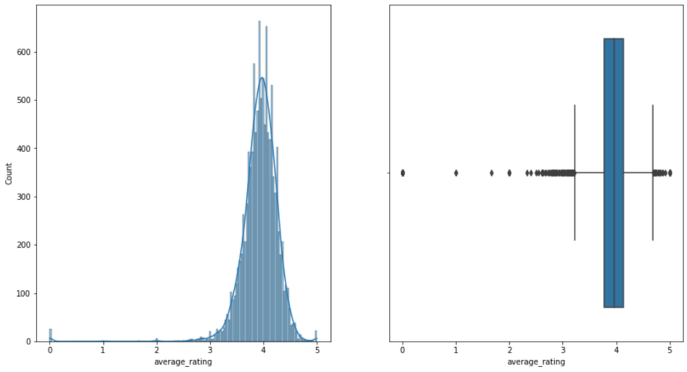

Figure 1 – ratings distribution

3.1 Renaming and splitting

I have started by reneaming the num_pages as there were a escape in the name and the next step was to retrieve the author name of the attribute authors. In fact the over name containing authors were illustrators/artist contributing to the publication. We have several well known example with a book like the Silmarillion of J.R.R tolkien.

So i have slit this column bas on character "/" to retrieve a the principal writer and i have created a new column Illustrator were values are either 0 or 1. 1 if the book has an illustrator 0 othewise.

3.2 Outliers

Numerical features as ratings_Count, text_reviews_count and num_pages contained a lot of outliers making hard to see their distribution with boxlot.

I choose to remove they outliers using the interquartil, the first quartile Q1 and the thid quartile Q3. For each of thes three features all value above Q3+1.5*IQR or below Q1-1.5IQR will be removed. In other hand it was also necessary to deals with incorrect values as books with a num_pages of zero, no reviews or rating. I choose to remove these observations as if no rating has been given the average_rating should be zero.

The last step with outliers was to remove values above 2500 as 75% of the values were below 1341.

4 Features engineering Selection

4.1 Features selection

I choose to use the identifiers BookId, isbn,isbn13 and the title,because because those values are unique. The authors names,the number of pages, the number of rating, the number of review, the publisher and the language seems to be the most logicals features to use (the kind could have also been a good features to use but we didn't have this information). A correlation matrix highlights a strong correlation between the ratings_count and the number of text review which makes sense but as the correlation was not equal or above 0.95 i choose to not drop any of these two features.

4.2 Features engineering

4.2.1 Authors and Publisher

These two are categorical values i use a labelEncoder to encode them into numerical. These two featues have no apparent order so the LabelEncoder seems to be the good choice.

4.2.2 Language_code

For the languages i choosed to transform them into dummies, although as for the authors and the publishers there is no apparent order between the different languages even though english is the most present. So every language will be transformed into a 1D array of 0 and 1, 1 if the language is present 0 otherwise.

4.3 Ratings_count,Text_reviews_count,num_pages

As we said before these 3 features are negatively skewed, a log transformation could have produced a better observation but for a regression we don't need the independant variable to be normally distributed.

In other hand a standarization of these values could help to interpret the results of our furure models. In fact, with a standarization all features would have relatively the same magnitude so we could discover the most important features to predict the average_ratings.

I dediced to use these features raw and standardized to compare the result and to enrich the discussion.

Here is the formula of a standardization :

$$X_{newi} = \frac{X_i - \mu}{\sigma} \tag{1}$$

where X_i is the value to transform, μ the mean and σ the standard deviation for a features. This result for a feature to have a mean of 0 and a standard deviation of 1.

5 Models

5.1 Chosen model

I have chosen 3 models to train and test, a simple linear regression, a random forest regression and a gradient-BoostingRegressor(ensemble model).

To evaluate the models as we are in a regression problem i have chosen 3 metrics :

The mean absolute error :

$MAE = \sum_{i=1}^{D} |y_i - y_{predictedi}|$

The mean scared error :

$MSE = \sum_{i=1}^{D} (y_i - y_{predictedi})^2$

the Root Mean Squared Error :

$RMSE = \sqrt{MSE}$

So the goal is to quantify the errors.

5.2 Results

I have splitted the date into a trainSet and a testSet with 80% of the data for the train. After evaluating the models with standarized features and raw features i could conclude that standarization has no effects on the results.The model that has achieved the best performance is the Random forest.

5.2.1 RandomForestResults

Mean Absolute Error (MAE) : 0.21253279073277487

Mean Squared Error (MSE) : 0.07588036054722806

Root Mean Squared Error (RMSE) : 0.27546390062443404

5.2.2 LinearRegessionResults

Mean Absolute Error (MAE) : 0.2214101973072093

Mean Squared Error (MSE) : 0.08118552114634263

Root Mean Squared Error (RMSE) : 0.28493073043521056

5.2.3 GradientBoostingResult

Mean Absolute Error (MAE) : 0.21520857635063131

Mean Squared Error (MSE) : 0.0780093336141369

Root Mean Squared Error (RMSE) : 0.2793015102253063

So we can say that we have some good models to predict a books rating. As the random forest has achieved the best performances i have chosen to compute features importance.

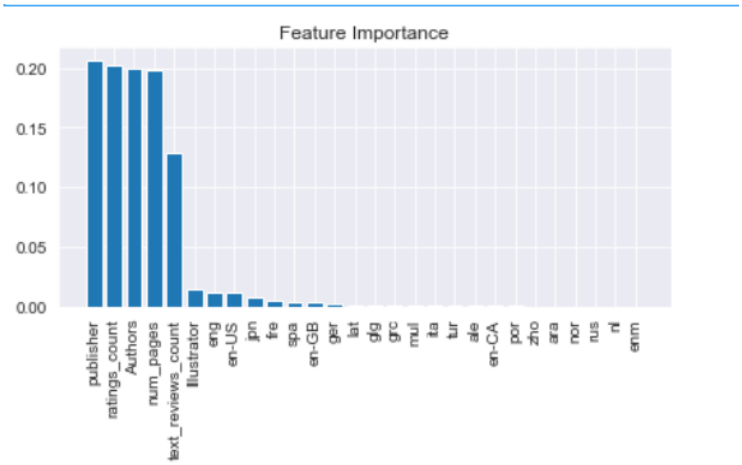

FIGURE 2 – Features importances

It seems that the most important features are the publisher, the author,the number of pages, the number of text review and the number of rating.

In the next section i will briefly explain a API that have built to predicted the average_rating of books with docker. The model used is a linear regression wich needs only the most important features mentioned previously.

6 Dockerization

For the purpose of future inference i have written an application which provides a several APIs. The application is implemented with flask and docker. The endpoint allowing to make predictions is $/training/train/model/predic$.To use the API you need to provide the author of the book, the number of pages, the number of ratings and the number of review.

The chosen model is inside volume and at the moment only the endpoint that makes predictions is fully operational.

The project is hosted at clik here.

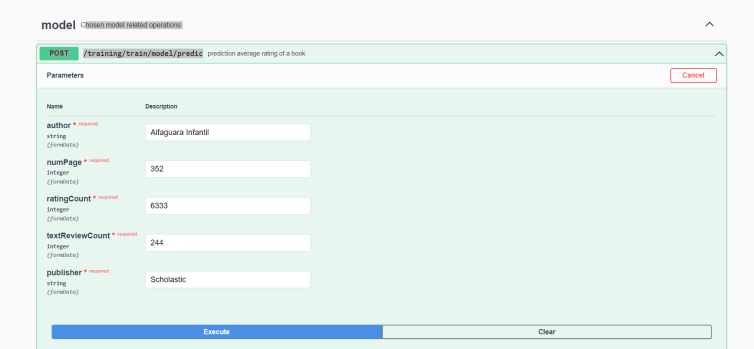

FIGURE 3 – API

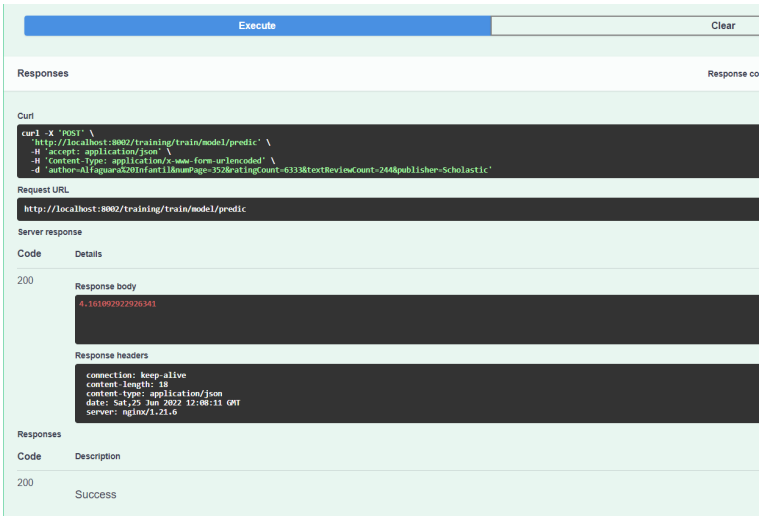

FIGURE 4 – API