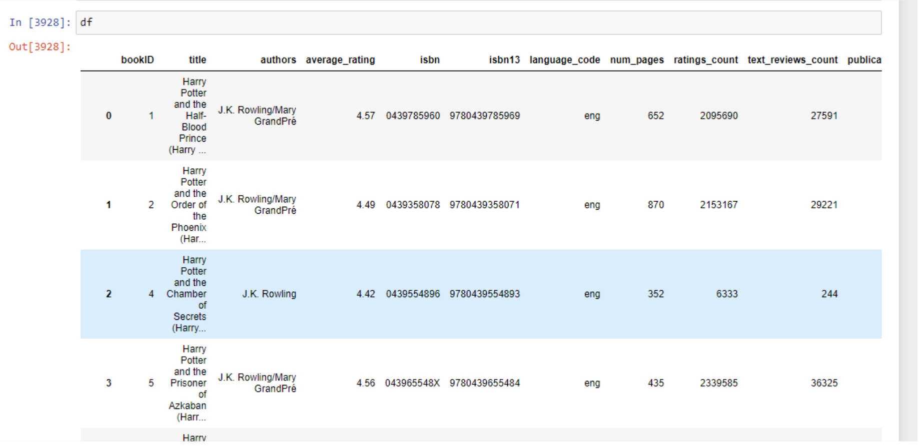
DA Python ML Lab Project Report: Goodreads Books Rating Prediction Model

With this project I tried to build a model to predict the average rating of books in a Goodreads Dataset I was provided with.

I committed this project on my github page : https://github.com/cquittat/goodreads

I began by importing the .csv dataset in a Jupyter notebook, and at first, I encountered some issues because comas lied in some cells of the CSV file, so I had to delete them by hand for 4 lines.
I also striped some spaces in the columns names to avoid problem calling them.

I didn't take the bookIds to do the index because to numbers were not following each other's, it seemed clearer to me.



```
In [3928]: df
```

Out[3928]:

	bookID	title	authors	average_rating	isbn	isbn13	language_code	num_pages	ratings_count	text_reviews_count	publica
0	1	Harry Potter and the Half-Blood Prince (Harry ...	J.K. Rowling/Mary GrandPré	4.57	0439785960	9780439785969	eng	652	2095690	27591	
1	2	Harry Potter and the Order of the Phoenix (Har...	J.K. Rowling/Mary GrandPré	4.49	0439358078	9780439358071	eng	870	2153167	29221	
2	4	Harry Potter and the Chamber of Secrets (Harry...	J.K. Rowling	4.42	0439554896	9780439554893	eng	352	6333	244	
3	5	Harry Potter and the Prisoner of Azkaban (Harr...	J.K. Rowling/Mary GrandPré	4.56	043965548X	9780439655484	eng	435	2339585	36325	
		Harry									

Then I started to explore the data, and though I didn't find NAs at first, I rapidly saw that some columns had "0" values typed in, so I checked column by column.

In the average_rating column I excluded all the "5" grades because they had very few rating_counts (less than 5), it seemed not relevant.
And after that I excluded all the "0" average_rating rows because they didn't have any ratings_count ("0" value), those were probably books that were never rated.

In the ratings_count column I saw many were still with a "0" value, it was weird because they had all a non-zero average_rating. As I couldn't find a reason to that, I excluded them from the dataset.

In the text_reviews_count column they were some “0” values, but as they had some ratings it seemed OK to me, it’s not illogical to have ratings but no text review.

Both for ratings and text reviews I found they were some very high values, checking with the top rows I found they were the same: they’re the most popular books (Twilight, The Alchemist, Harry potter…), so I didn’t treat them as outliers and kept them.

In the num_pages column I first saw that we had some books with a large number of pages (more than 1000 seems a lot, it’s fairly rare to have so many pages).
But looking more closely I found that those books are for the most part sets of books, so it’s logical, no need to exclude them.

But then I still had one problem: 75 rows did have a “0” value for the num_pages, probably the information weren’t known at the time the dataset was made. I also saw 194 rows with num_pages lower than 20, which was.
So I decided to replace the “0”and “less than 20” values by the mean of all the books pages (around 337 pages) for a better training of the models.

Then I looked at the different languages of the books, and most of them didn’t have more than 50 books in the dataset, so I preferred to exclude them and just keep the 7 most representative languages.

For the publication_date column I needed first to convert the column into date format, I also excluded two rows because their dates were generating errors.
I also showed the 10 years with the most books published.

I added some quick analysis to respond to a few questions I asked myself :

- What are the 10 most represented Publishers in the dataset ?
- What are the 10 most represented Authors in the dataset ?
- In average which language has the books with the more pages ? (I always thought French books had more pages, but I would need to compare the same books in different languages to do so ^^)

Then came the moment to prepare the dataset for Machine Learning.

First I transformed language_codes in dummies variables to extant the data models could process.

And for the date part I only kept the years as integers, then they could be processed by the model.

At last I dropped the unuseful columns (text columns, columns that were processed into numbers, “isbn” columns).

Then I drew the correlation matrix for this processed dataset, I could see a few correlations, though there are not much on the average_rating line : only the num_pages columns seems to have a faint correlation with it.
Then I drew some scatterplots for the relation between average ratings and: ratings_count, text_reviews_count and num_pages, who all seem to be positively related to it.

After all I began the Machine Learning part.

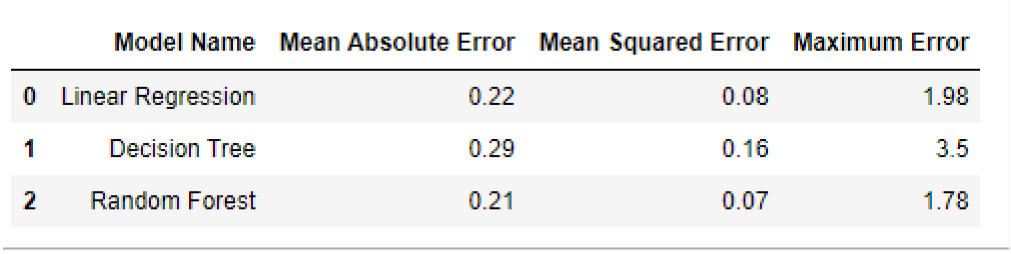
Because we were trying to predict float values the best to do was to load regression models.

I tried Linear Regression, Decision Tree and Random Forest from the scikit learn library.

For evaluating those models, I choosed:

- MAE / Mean Absolute error : to have a simple and intuitive metric
- MSE / Mean Squared error : to have a better understanding of the large errors
- ME /Maximum Error : this one is less relevant but it was to know what was the largest error in the predictions

Then we can compare the three models and though all three did a fairly great job, the Random Forest one is clearly the best :



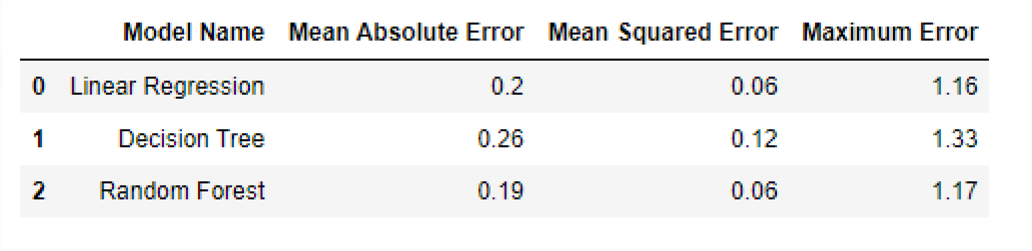
	Model Name	Mean Absolute Error	Mean Squared Error	Maximum Error
0	Linear Regression	0.22	0.08	1.98
1	Decision Tree	0.29	0.16	3.5
2	Random Forest	0.21	0.07	1.78

Then the last question: how to improve those results?

We saw above that ratings_counts, text_reviews_count and num_pages were related to average_ratings: the higher they were, they higher was the average rating, so it seemed unuseful to delete the higher rows for these.

But there is one thing to experiment: the low rated books. It's highly possible that books with less ratings don't have a significant average_ratings, I deleted them from the dataset.

Then I redid the Machine learning steps and the results are a little better, again the random forest seems to be the best model:



	Model Name	Mean Absolute Error	Mean Squared Error	Maximum Error
0	Linear Regression	0.2	0.06	1.16
1	Decision Tree	0.26	0.12	1.33
2	Random Forest	0.19	0.06	1.17