Project Report: Book Rating Prediction Model

Python Machine Learning Labs

Ilias Khattab

Data Cleaning

In order to analyze and train a machine learning algorithm, we must first clean the data. This was done through the following steps:

1. Removed the space before the num_pages column.
2. Set the bookID as the index.
3. Checked for any null values.
4. Checked for any duplicates.
5. Split the "authors" column into two new separate columns, namely "AUTHOR" and "ILLUSTRATOR".
6. Removed outliers from the dataset by setting boundaries for values such as num_pages (e.g. it needs to be above zero).
7. Transformed the variables: they were skewed and did not have a Gaussian distribution.

Exploratory Data Analysis

Once the data has been cleaned, we can move on to exploratory data analysis, where we will obtain an overview of the data and understand how it is distributed.

Through plotting the main variables, we found out that most books have an average of 200-500 pages. Most reviews are contained between 10 to 50 characters. Most books have a rating count ranging from 100 to 700. Most books in this database are written in English, followed by Spanish, French, German, and lastly, Japanese. A Shapiro testing outputs a p-value below 0.05, so we can say that the average rating is normally distributed. More insights can be found in the Jupiter Notebook, including the book with the most ratings, as well as the publisher, author, and illustrator, with the greatest number of books.

Feature Engineering and Selection

In order to predict the rating, we need to select variables that may have an impact on the rating itself. Certain variables are irrelevant to it, and accordingly, ISBN and ID were not selected. The following variables have been kept:

- Num Pages
- Text Reviews Count
- Ratings Count

Following our selection, we have split the data in two parts. 80% of the data became the training group and 20% became the testing group.

We inputted the training data into three different models in order to train and evaluate them:

- Linear Regression is an equation with the features above.
- Ridge Regression is an equation with a penalization system.
- Deep learning used with 3 entry values (num page, text reviews count and rating) 1 Exit value (the average rating) with a linear activation, 50 epochs with a learning rate 0,001 and an Adam optimizer

Model Training and Evaluation:

We chose the **Mean absolute error:**

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n}$$

Yi is the real value and x is the predicted value.

We chose the **Mean squared error:**

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

Yi is the real value and ÿi is the predicted value

Root Mean Squared Error (RMSE): is the squared root of the MSE above.

Results

1. Linear Regression:

We used a power transform because we wanted the model to learn from a standardized data

Mean Absolute Error (MAE): 0.22351345163384315
Mean Squared Error (MSE): 0.0812107921003558
Root Mean Squared Error (RMSE): 0.28497507277015616

2. Ridge Regression

Mean Absolute Error (MAE): 0.2254637821041687
Mean Squared Error (MSE): 0.0822989062463118
Root Mean Squared Error (RMSE): 0.28687785945644495

3. Deep learning:

Mean Absolute Error (MAE): 0.2254853824392637
Mean Squared Error (MSE): 0.08239130400185607
Root Mean Squared Error (RMSE): 0.28703885451599764

Conclusion

The best model is the linear regression, which has the lowest error values. The models are on average very good since the mean absolute error of 0,22 is very decent. The Mean squared Error are under 0,0

From the results we can deduce that the three variables chosen for the prediction were probably the best.

To go further:

We didn’t use the publisher/author/illustrator’s name, since those may be converted to a numerical value with a label encoder. These variables might have had an impact on the average rating predicted.