Develop an end-to-end Machine Learning Pipeline

Course: Python Machine Learning Labs

Instructor: Assan Sanogo

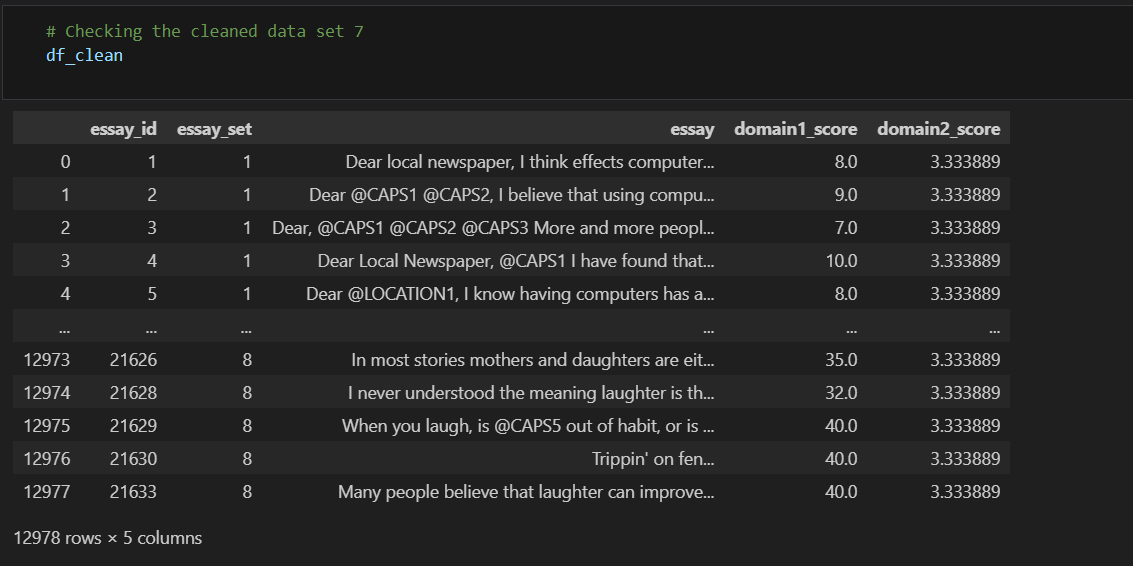
Professor: Hanna Abi Akl

Students: Yaakoub El Guezdar, Kateryna Draganova

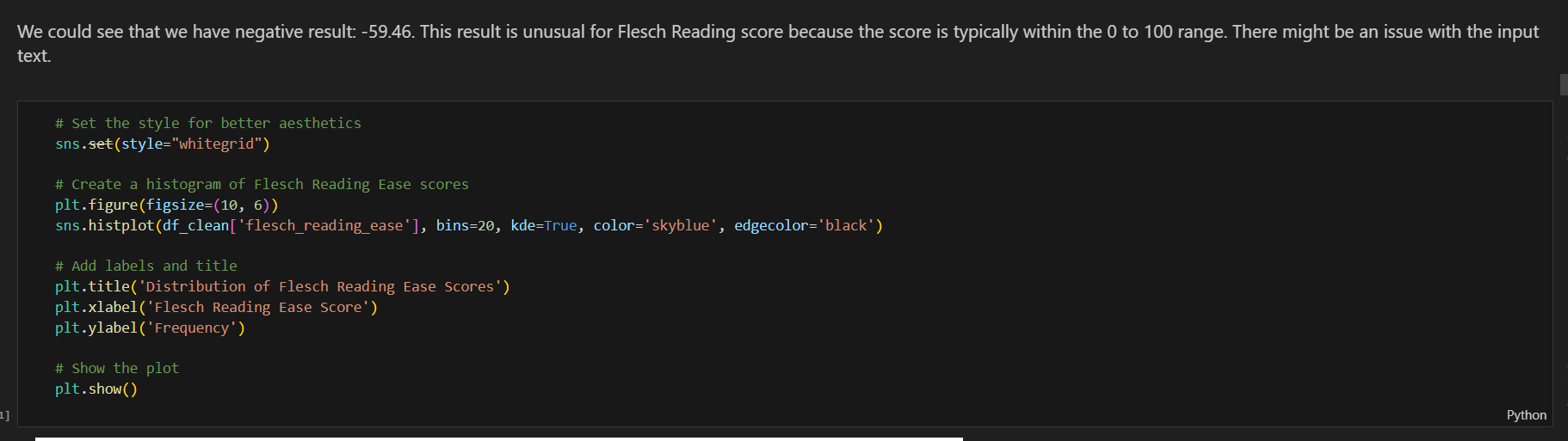
Data set source: <https://www.transferxl.com/download/08vSFcz3B7fXPr>

The code includes: Data exploration which includes:

1. Data Extraction from data set source.
2. Exploratory Data Analysis (EDA). We filtered data and check each type of essays separately, verified for NaN values and dropped those columns which had them. Then we discovered properties and features which could be used in training of the model:



1. Choosing features: essays length (number of words); longer essays may imply a thorough examination, offering a comprehensive and detailed analysis. On the other hand, shorter essays might provide brevity but may sacrifice a nuanced exploration of the subject matter.
2. Extracting tags in the essays and printing the tagged words and their categorization; counting the **frequency of the tagged words** from all categories in each essay**. tags\_count** isn’t suitable for our ML model, but it represents total number of tags in each essay.
3. **Ratio of sentences to paragraphs**. The sentence-to-paragraph ratio in essay writing offers insights into writing style and content. A higher ratio, with more sentences per paragraph, suggests a detailed and complex style. However, it may also indicate a preference for expressing ideas in smaller, more digestible chunks, potentially enhancing clarity and facilitating better communication with the reader. Such essays often feature a mix of shorter and longer paragraphs, contributing to a varied and engaging writing style.
4. **Sentence length variation**. This variation, mixing short and long sentences, boost readability by preventing monotony and tailoring content complexity to the audience. This stylistic approach adds expressiveness, using short sentences for emphasis and longer ones for detailed explanations. The sentence length variation maintains reader engagement, ensuring a smooth flow of ideas in the narrative or argument. Analysing sentence length offers insights into analysis depth, target audience considerations, and areas for editing. In persuasive writing, more sentences per paragraph suggest detailed arguments, while narrative and expository writing may focus on storytelling or concise explanations with a balanced or slightly lower sentence ratio.
5. **Sentence length variation**. This variation, mixing short and long sentences, boost readability by preventing monotony and tailoring content complexity to the audience. This stylistic approach adds expressiveness, using short sentences for emphasis and longer ones for detailed explanations. The sentence length variation maintains reader engagement, ensuring a smooth flow of ideas in the narrative or argument. Analysing sentence length offers insights into analysis depth, target audience considerations, and areas for editing. In persuasive writing, more sentences per paragraph suggest detailed arguments, while narrative and expository writing may focus on storytelling or concise explanations with a balanced or slightly lower sentence ratio.
6. **Transition words**. Transition words are vital for essay coherence, guiding readers through thoughts and arguments, enhancing overall structure and clarity. They facilitate smooth transitions between essay parts like introductions, paragraphs, and conclusions. In addition, they explain the relationships between concepts, whether through cause and effect, contrast, comparison or other relationships, and thereby explain the accuracy of the writing. Skilful use of transition words not only facilitates understanding, but also reflects more advanced writing skills, showing the writer's ability to logically connect ideas and create a complex essay.
7. **Corrected type token ration** (**CTTR**). CTTR is a measure of lexical diversity in a text. It calculates the ratio of unique words to the square root of twice the total number of words divided by the number of unique words. This ratio is used to estimate the lexical richness or diversity of the text. A higher CTTR indicates greater lexical diversity, suggesting a wider range of vocabulary usage.
8. **Number of sentences**. By quantifying the sentence count, we gain insights into the structural complexity and coherence of the essay's content. A higher number of sentences may indicate detailed exposition, complex argumentation, or nuanced storytelling, while a lower count might suggest concise writing or a focused narrative.
9. **Word frequencies** + **Most repeated words.** Analyzing word frequency helps us understand how diverse and varied the vocabulary is in each essay. Examining frequent word usage reveals the richness of language. Comparing this across essays helps identify common patterns or trends. Word frequency indicates the primary topics or focal points of the essays. High-frequency words signal the importance of specific themes in the content. We can also use word frequency to see how well-written the essays are. If there are lots of different words used, it suggests the writing is more interesting. But if the same words keep appearing, it might mean the writing is repetitive or not very engaging.
10. **Flesch-reading Ease.** At this step we started to calculate the readability of the essays. The Flesch Reading Ease score assesses text readability by considering linguistic complexity. It provides a numerical indication of how easy or challenging a piece of writing is to comprehend. Factors like average sentence length and syllables per word contribute to the score. A higher Flesch Reading Ease score implies easier readability, characterized by simpler sentences and words, whereas a lower score suggests more complex language and potentially harder comprehension.



**The resulting score** could be interpreted as:

90-100: Very easy (e.g., 5th-grade level)

80-89: Easy (e.g., 6th-grade level)

70-79: Fairly easy (e.g., 7th-grade level)

60-69: Standard (e.g., 8th and 9th-grade level)

50-59: Fairly difficult (e.g., 10th-12th-grade level)

30-49: Difficult (e.g., college level)

0-29: Very difficult (e.g., graduate level)

1. **Gunning fog index**. By examining the average words per sentence and the percentage of complex words, the index evaluates the passage's complexity. It offers an estimated grade level necessary for comprehension, with higher scores indicating greater difficulty. A higher Gunning Fog Index score denotes more challenging reading material, whereas a lower score signifies easier readability.

The resulting score could be interpreted as:

6-8: Very easy (e.g., 6th-8th-grade level)

8-12: Easy (e.g., 9th-12th-grade level)

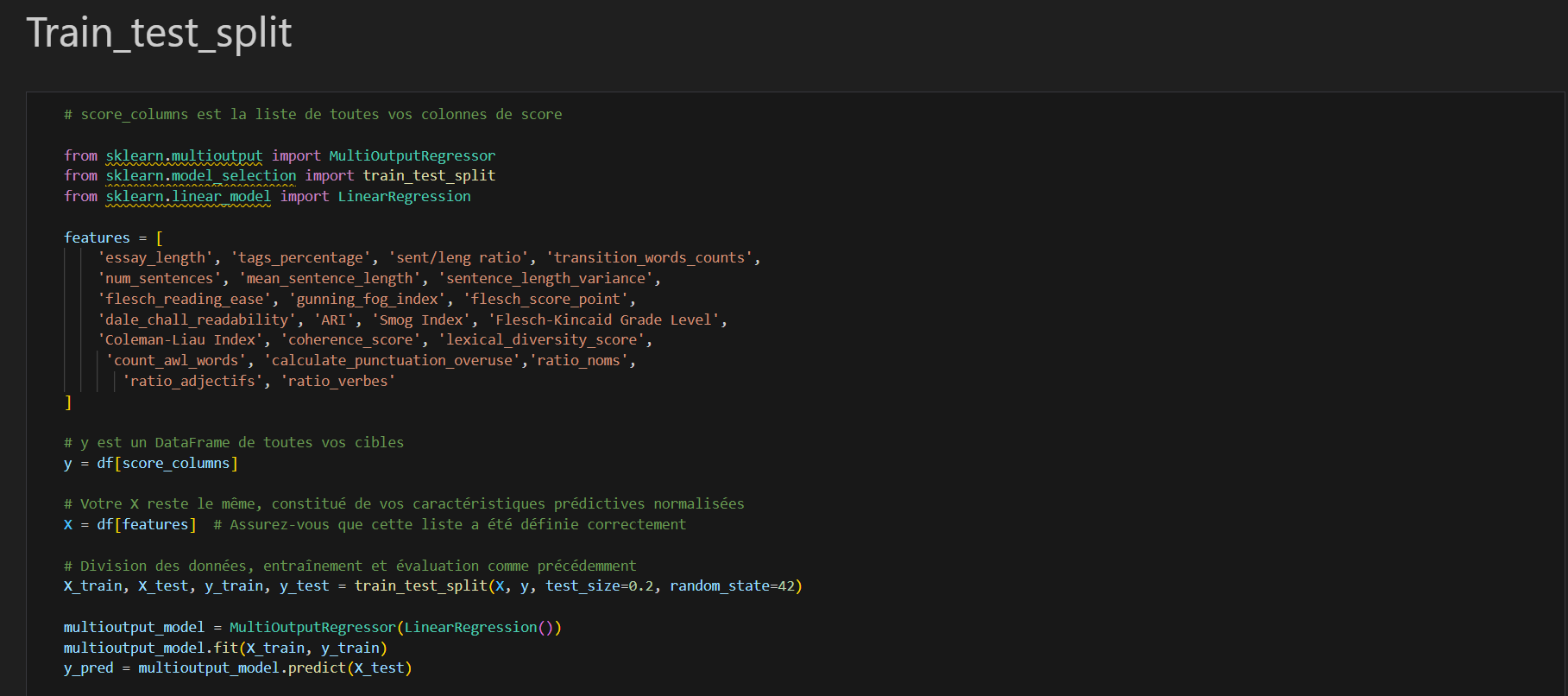
12-16: Fairly difficult (e.g., some college level)

16-18: Difficult (e.g., college graduate level)

18 and above: Very difficult (e.g., advanced graduate level)

1. **Misspelling rate**. Then we checked the misspelling rate in each essay. Misspelling refers to the incorrect spelling of words in written text. It occurs when a word is written with letters that do not match the accepted spelling of that word in a particular language. Misspellings can occur due to various reasons, such as typographical errors, lack of knowledge about the correct spelling, or confusion between similar-looking or similar-sounding words. **misspelled\_**words and **missprlling\_score** were dropped.
2. Analysing frequencyof **Adjectives, Adverbs, Nouns and Verbs**. Analysing how often different types of words like describing words (adjectives), action words (verbs), and others appear in an essay helps us understand how varied and interesting the writing is. If there are lots of describing words, it might mean the writing is colourful and vivid. More action words could show the writing is dynamic and engaging. By looking at these word types, we can see how well the author uses language to make their writing lively and engaging.
3. Machine learning models which we used:
4. "We developed this code to preprocess and enrich a dataset from an Excel file (valid\_set.xlsx). Our goal was to extract meaningful features and insights from essays in the dataset. The code performs extensive feature engineering, including calculations for essay length, tag counts, readability scores, misspelling statistics, and various linguistic features. We utilized natural language processing tools such as spaCy for text analysis. The resulting Valid\_Set DataFrame is now enriched with a diverse set of features, making it suitable for in-depth data analysis or machine learning applications.
5. "We utilized the `MinMaxScaler` from scikit-learn to preprocess the features in the `Valid\_Set` DataFrame. Our objective was to normalize specific predictive features, including essay length, tag percentages, readability scores, and various linguistic attributes. By applying the Min-Max scaling technique, we transformed these features to a standardized range, ensuring consistent and comparable magnitudes across the dataset. This normalization is crucial for maintaining numerical stability and improving the performance of machine learning models that rely on these features. The resulting `Valid\_Set` DataFrame now contains normalized features, setting the stage for more robust and accurate predictive modeling using techniques like the `MultiOutputRegressor` from scikit-learn."
6. "We implemented a multi-output regression model using scikit-learn to predict multiple target scores based on a set of features. The predictive features, including essay length, tag percentages, readability scores, and linguistic attributes, were previously normalized using the Min-Max scaling technique. The target scores are contained in the `score\_columns` list, and the feature matrix is represented by `X`, while the target matrix is denoted as `y`.
7. We split the dataset into training and testing sets (80% training, 20% testing) using the `train\_test\_split` function. Subsequently, a `MultiOutputRegressor` was initialized with a linear regression model as the base estimator. The model was trained on the training data (`X\_train` and `y\_train`). Following training, predictions were generated for the test set (`X\_test`), and the results were stored in the `y\_pred` variable.

This approach enables us to simultaneously predict multiple scores, making it particularly useful for tasks where the targets are correlated. The linear regression model serves as the underlying predictor, and the multi-output wrapper extends its application to handle multiple target variables efficiently. Overall, this strategy forms a foundation for predicting a set of scores based on the specified features in a streamlined and interpretable manner."



1. "We employed various regression models from scikit-learn to predict multiple target scores based on the provided features. The models chosen for this task include Linear Regression, Random Forest Regressor, Support Vector Regression (SVR), and Multi-layer Perceptron (MLP).

To ensure optimal performance, SVR and MLP models were incorporated into pipelines with StandardScaler for feature normalization. This preprocessing step is often beneficial for SVR and MLP models, as they tend to perform better with normalized data.

For Random Forest and Linear Regression models, normalization was not necessary, and thus they were directly employed as MultiOutputRegressor instances without additional preprocessing steps.

A dictionary named `models` was created to facilitate the iteration over different models. Each model was trained on the training data (`X\_train` and `y\_train`), and predictions were generated for the test set (`X\_test`). Subsequently, Mean Squared Error (MSE) and R-squared (R^2) scores were calculated and displayed for each model to evaluate their performance.

This comprehensive approach allowed us to compare the effectiveness of different regression models in predicting multiple target scores, providing valuable insights into their respective strengths and weaknesses."

1. We used two metrics to evaluate the performance of the models:

- Mean Squared Error (MSE): which measures the quality of an estimator by calculating the average of the squares of the errors, that is, the average squared differences between the estimated values and the actual values.

- R² (Coefficient of Determination): which provides a measure of how well a regression model predicts. It represents the proportion of the variance for a dependent variable that's explained by a linear model.

Based on the results, the assessment of each model:

Linear Regression MSE: 16.819

Linear Regression R^2: 0.328

RandomForestRegressor MSE: 10.283

RandomForestRegressor R^2: 0.411

Support Vector Regression MSE: 13.486

Support Vector Regression R^2: 0.401

To choose the best model, we typically look for a low MSE and a high R² (as close to 1 as possible). A negative R² indicates that the model fits very poorly.

From these results:

- The \*\*Multi-layer Perceptron\*\* has the lowest MSE, indicating that on average, it made the closest predictions to the true values on the test data.

- However, its R² is negative, suggesting that the model does not capture the variance of the data well and might perform worse than a simple average.

- \*\*Linear Regression\*\* has a positive R², although it's low, which suggests it can capture some variance in the data, but with a slightly higher MSE than the multi-layer perceptron.

- \*\*Support Vector Regression\*\* has the poorest performance in terms of R², indicating a very poor model fit.

Thus, based on these metrics, \*\*RandomForestRegressor\*\* appears to be the best model among the three because it has a positive R² and a relatively low MSE. It's important to note that even though the multi-layer perceptron has the lowest MSE, having a negative R² suggests there are issues with how the model generalizes to the data, which could be due to overfitting or other issues in the model setup.