LINEAR QUANTILE REGRESSION

This code appears to be a **linear quantile regression** model for predicting the length of a product given its type. Here is a step-by-step explanation of the code:

- The necessary libraries are imported: pandas for data manipulation, numpy for numerical operations, statsmodels for statistical modeling, and mean_absolute_percentage_error from sklearn.metrics for evaluating the model's performance.The drive.mount() function is used to mount Google Drive in Google Colab.

- The test data is loaded into a pandas DataFrame and the index is set to the PRODUCT_ID column. The BULLET_POINTS, DESCRIPTION, and TITLE columns are dropped because they are not needed for the model.

- The train data is loaded into a pandas DataFrame and the index is set to the PRODUCT_ID column. The BULLET_POINTS, DESCRIPTION, and TITLE columns are dropped because they are not needed for the model.

- The necessary features for the model are selected: PRODUCT_TYPE_ID for x_train and x_test, and PRODUCT_LENGTH for y_train and y_test.

- A linear quantile regression model is created using statsmodels.formula.api.smf.quantreg(). The formula "PRODUCT_LENGTH ~ PRODUCT_TYPE_ID" specifies the dependent variable and independent variable(s) used in the model.

- The fit() method is called on the model object to train the model on the training data.

- The predict() method is called on the model object to generate predictions on the test data.The predictions are stored in a pandas DataFrame called "pred" with columns "PRODUCT_ID" and "PRODUCT".

- The "pred" DataFrame is exported as a CSV file called "submission q2 model.csv".

- The model's performance is evaluated using the mean_absolute_percentage_error function and the score is printed to the console.

The tools used are pandas, numpy, statsmodels, and mean_absolute_percentage_error from sklearn.metrics.