ML Regression Assignment

This report deals with different regression techniques used to predict the transcoding time of a video based on several parameters of both, input and output compression format. The dataset contains the information of 12000 videos with the respective target variable.

Data import

First of all, open source libraries, pandas, numpy, seaborn, and matplotlib.pyplot, have been imported as pd, np, sns, and plt respectively. Then, the dataset “model.csv” has been uploaded. Thanks to the `shape` function the dataset’s dimensions, equal to `(12000, 23)`, are shown.

Exploratory Data Analysis

The second step is the data exploration, in which duplicated rows and not a numbers have been searched into the dataset. In particular, none not a numbers and duplicated rows have been found in the dataset.

Split categorical and numerical variables

During the creation of a model, one of the most important thing to do is the feature selection in order to define which subset of variables could be useful to make the prediction. Thanks to the “`dtypes`” function is possible to recognize the type (categorical or numerical) of each variable.
Furthermore, all the categorical variables have been added to `df_categorical` while the numerical ones have been put in to the `df_numerical`. Then, the empty *b_size* variable has been dropped.

Categorical Data

In order to analyse categorical data, the catplot has been used to compare the box plots related to each categorical variable. For instance, considering the *codec* variable the box plots obtained are linked to the different coding standard used for the video.

Numerical Data

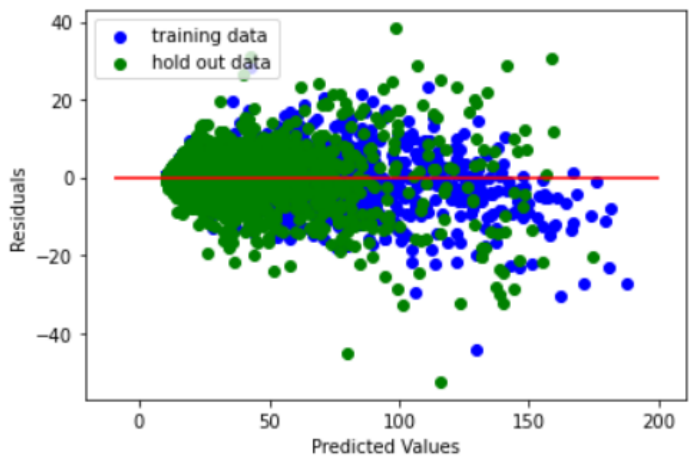
By the same token, numerical data have been explored using the histogram. Thanks to the Heat Map in Seaborn it could be seen the correlation between numerical variables, while using the Pair Plot it could be visualized the different scatterplot, which show the relationship between the features and the response. Analysing the Heat Map, the variables which are highly correlated have been dropped because it means that they are redundant. For these reasons, *height, p, p_size* and *o_height* have not been considered.

Standardize

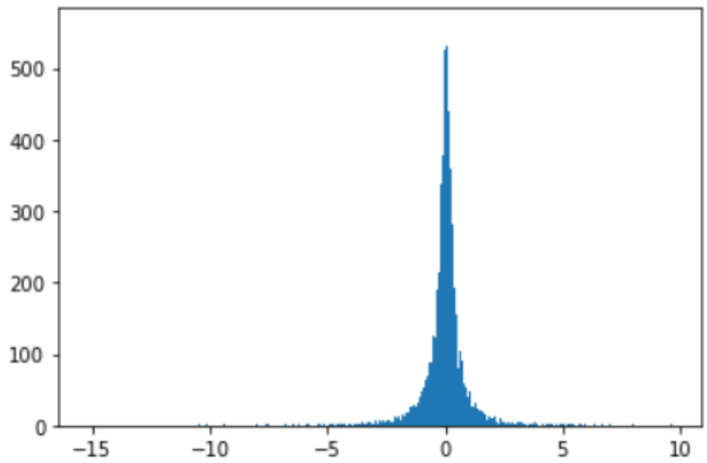
All the variables have been put in a standard form in order to make comparable the different values. It has been done because some measures could be in a different scale or in a different unit of measurement. As a result, the variable `X` has been created thanks to the concatenation between the two variables (`dummies` and `X_numerical`) which contain the categorical and numerical chosen variables.

Models

Before choosing the best model, the two sets (train and test sets) have been split. A stratify selection has been used with a `test_size` equal to 0.30. Various approaches have been proposed to solve this problem, but it can be seen from the data that the best model is the Random Forest Regression because MAE train and MAE test values are quite small and similar which means that there isn't overfitting. The requisites for having a very good model are: constant variance, mean value near to zero and a normally distribution of the errors around zero. Furthermore, the distribution should be random, otherwise it could be suspicious. Using the scatterplot, it could be seen the distribution of the residuals of the predicted values. It seems to be random and equally distributed around zero.

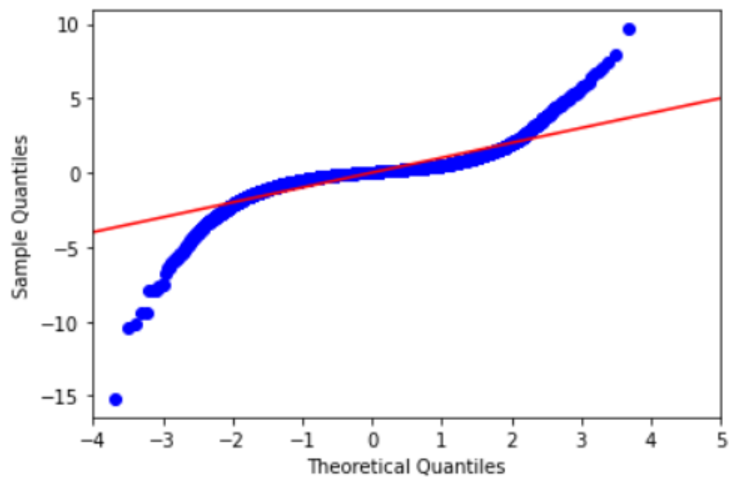


Thanks to the histogram, it could be seen that the distribution of the errors is really good and the mean is on zero.



But there are several methods that can prove it. For example, the qq plot, the Kolmogorov-Smirnov Test and the D'Agostino Test. In particular, the last two methods return the p-value. In both cases, the p-value is very small and so the null hypothesis could be rejected. But there isn't enough statistical evidence to say that the distribution is normal.

- Kolmogorov-Smirnov Test
  `KstestResult(statistic=0.15236297214544703, pvalue=1.0003039809388876e-170)`
- D'Agostino Test
  `NormaltestResult(statistic=2131.579448705327, pvalue=0.0)`
- qq plot
  Analysing the qq plot, it could be seen that there is a problem on the extremes of the graph, while from -2 to +2 the curve trend is quite normal.



In brief, shown below the results of each model.

- Linear Regression

```
***GRIDSEARCH RESULTS***
Best score: -9.653986 using {}

MAE  train 9.601    test 09.812
MSE  train 203.465 test 210.396
RMSE train 14.264   test 14.505
r2   train 0.628     test 0.619
```

- Ridge Regression

```
***GRIDSEARCH RESULTS***
Best score: -9.341899 using {'alpha': 0.1, 'normalize': True}

MAE  train 9.301    test 09.561
MSE  train 220.919 test 228.616
RMSE train 14.863   test 15.120
r2   train 0.597     test 0.586
```

- Lasso Regression

```
***GRIDSEARCH RESULTS***
Best score: -9.392168 using {'alpha': 0.01, 'normalize': True}

MAE  train 9.361    test 09.536
MSE  train 226.712 test 232.335
RMSE train 15.057   test 15.243
r2   train 0.586     test 0.579
```

- KNeighbors Regression

```
***GRIDSEARCH RESULTS***
Best score: -9.224840 using {'n_neighbors': 20, 'p': 2}

MAE  train 8.235    test 08.894
MSE  train 174.776 test 198.874
RMSE train 13.220   test 14.102
r2   train 0.681     test 0.640
```

- Decision Tree Regression

```
***GRIDSEARCH RESULTS***
Best score: -5.259488 using {'max_depth': 7, 'min_samples_leaf': 5}

MAE  train 4.790    test 05.024
MSE  train 64.320 test 71.861
RMSE train 8.020   test 8.477
r2   train 0.883     test 0.870
```

- Random Forest Regression

```
***GRIDSEARCH RESULTS***
Best score: -4.233209 using {'criterion': 'mse', 'min_samples_leaf': 10, 'n_estimators': 100, 'random_state': 42}

MAE  train 3.114    test 03.708
MSE  train 30.371 test 40.634
RMSE train 5.511   test 6.374
r2   train 0.945     test 0.926
```

- MLP Regression

```
***GRIDSEARCH RESULTS***
Best score: -5.476881 using {'alpha': 0.01, 'batch_size': 20, 'hidden_layer_sizes': (10, 5), 'learning_rate': 'constant', 'max_
iter': 1000, 'solver': 'sgd'}

MAE  train 5.033    test 05.320
MSE  train 69.584 test 78.966
RMSE train 8.342   test 8.886
r2   train 0.873     test 0.857
```

- AdaBoost Regression

```
***GRIDSEARCH RESULTS***
Best score: -8.802579 using {'learning_rate': 0.5, 'loss': 'linear', 'n_estimators': 5, 'random_state': 0}

MAE  train 9.646    test 09.663
MSE  train 197.066 test 194.978
RMSE train 14.038   test 13.963
r2   train 0.640     test 0.647
```

- Gradient Boosting Regression

```
***GRIDSEARCH RESULTS***
Best score: -7.660828 using {'learning_rate': 1, 'loss': 'lad', 'max_depth': 2, 'n_estimators': 10, 'random_state': 0}

MAE  train 7.964    test 08.155
MSE  train 238.600 test 245.600
RMSE train 15.447   test 15.672
r2   train 0.564     test 0.555
```