Exploratory Data Analysis Report

Soudabeh Bolouri

Introduction:

In this exercise we will have a look at the raw data before doing any machine learning. The objective is to familiarize yourself with the data, see if there are any potential problems or obstacles to applying machine learning, and if and what preprocessing may be helpful. We want to explore the distribution of selected features within the "winequality-white.csv" dataset before and after applying a normalization technique using the *Logarithmic Transformation*. We finally showed the distribution of feature values before and after preprocessing.

Dataset Description:

The dataset that we utilized in this exercise is the "Wine Quality" dataset, precisely the "white" wine version. It contains data on different features of white wines, including fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, PH, sulphates, and alcohol. The dataset includes a totality of 4,898 rows and 12 features. The "quality" of the wine is the target variable, which we desire to predict. Also, we did not find any missing values in the dataset.

Experimental Setup:

We set up the experiment as follows using the Python programming language:

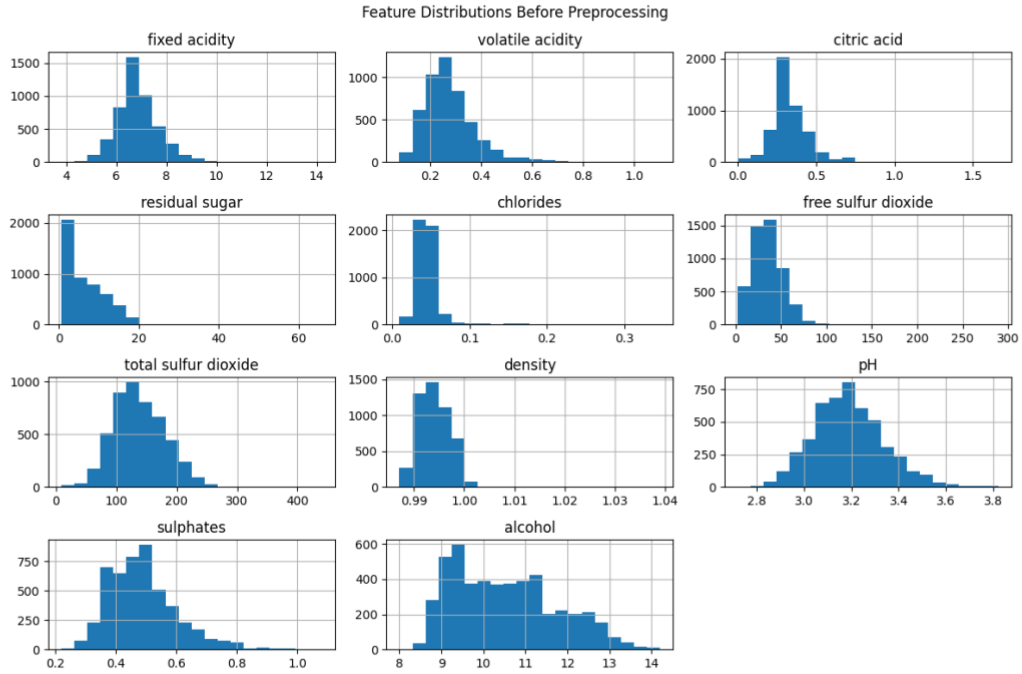
First, we visualized the distributions of the features before preprocessing using histograms. The histograms illustrate the spread and frequency distribution of each feature's values across the dataset.

Then we performed a logarithmic transformation on all features except the target variable 'quality'. Combining Pandas' apply function with NumPy's log function produces this transformation. To avoid taking the logarithm of zero or negative values, a lambda function, **lambda x: np.log(x + 1)**, is used; this function computes the natural logarithm of each value in the selected features after adding 1.

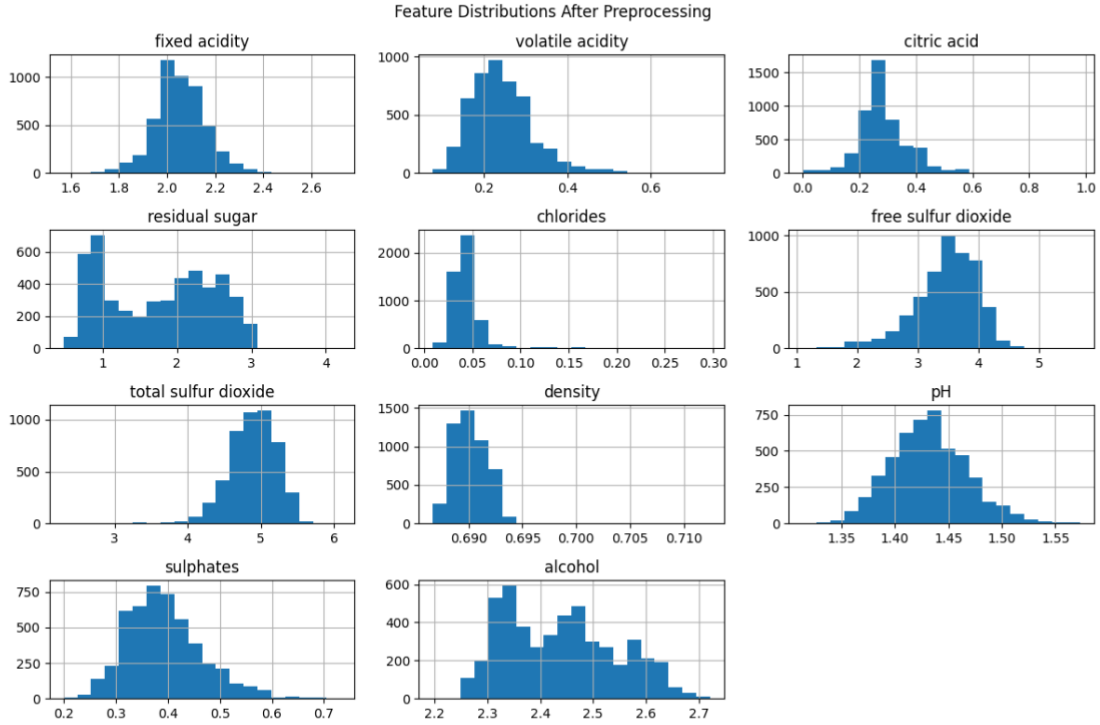
After applying the logarithmic transformation to the features, histograms are created again for these transformed features.

Results:

In the first histogram, the original distributions of the features were displayed.



Distribution of features after applying the logarithmic transformation:



Using this method, we can change the shape of the distributions, particularly for features with a wide range of values or that are heavily skewed, making it easier to model the data.

References:

[1] https://www.analyticsvidhya.com/blog/2020/04/feature-scaling-machine-learning-normalization-standardization/#

[2] https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MinMaxScaler.html

[3] https://www.w3schools.com/python/matplotlib_histograms.asp

[4] https://medium.com/@kyawsawhtoon/log-transformation-purpose-and-interpretation-9444b4b049c9