Title: Air Quality Prediction

The Dataset is taken from Kaggle(https://www.kaggle.com/nareshbhat/air-quality-analysis-eda-and-classification). The dataset consists of 26219 entries indexed from 0 to 26218. There are total 16 columns.

SimpleImputer is a scikit-learn class which is helpful in handling the missing data in the predictive model dataset. It replaces the NaN values with a specified placeholder.

It is implemented by the use of the SimpleImputer() method

missing_values : The missing_values placeholder which has to be imputed. By default is NaN

stategy : The data which will replace the NaN values from the dataset. The strategy argument can take the values – 'mean'(default), 'median', 'most_frequent' and 'constant'.

fill_value : The constant value to be given to the NaN data using the constant strategy.

fit_transform() and another one is transform(). Both are the methods of
class sklearn.preprocessing.StandardScaler() and used almost together while scaling or standardizing our training and test data.

The fit method is calculating the mean and variance of each of the features present in our data. The transform method is transforming all the features using the respective mean and variance.

transform()
Using the transform method we can use the same mean and variance as it is calculated from our training data to transform our test data.

Label Encoding refers to converting the labels into numeric form so as to convert it into the machine-readable form. Machine learning algorithms can then decide in a better way on how those labels must be operated. It is an important pre-processing step for the structured dataset in supervised learning.

Limitation of label Encoding
Label encoding convert the data in machine readable form, but it assigns a

unique number(starting from 0) to each class of data. This may lead to the generation of priority issue in training of data sets. A label with high value may be considered to have high priority than a label having lower value.

The dependent variable is the Airquality columns. There are 6 different classes of Airquality namely Good, Poor, Moderate, Satisfactory, severe and very poor

Moderate(1): 5450,
Satisfactory(3): 4808,
Poor(2): 1867,
Very Poor(5): 1618,
Severe(4): 951,
Good(0): 661

the distribution of the classes is uneven. To make the dataset distribution even, SMOTE technique is used. The Synthetic Minority Over-sampling Technique (SMOTE) is an oversampling approach that creates synthetic minority class samples. SMOTE first selects a minority class instance a at random and finds its k nearest minority class neighbors.

SMOTE

- Synthetic Minority Oversampling Technique

Creates new "Synthetic" observations

- SMOTE Process

- Identify the feature vector and its nearest neighbour

- Take the difference between the two

- Multiply the difference with a random number between 0 and 1

- Identify a new point on the line segment by adding the random number to feature vector

- Repeat the process for identified feature vectors

Moderate(1): 5450,
Satisfactory(3): 5450,
Poor(2): 5450,
Very Poor(5): 5450,

Severe(4): 5450,
Good(0): 5450

Dealing with Imbalanced Dataset

- Presence of minority class in the dataset

Challenges related Imbalanced Dataset

- Biased predictions

- Misleading accuracy

- Some Examples

- Credit card frauds

- Rare diseases diagnosis

Re-Sample the Dataset

Balance the classes by Increasing minority or decreasing majority

- Random Under-Sampling

- Randomly remove majority class observations

- Helps balance the dataset

- Discarded observations could have important information

. May lead to bias

- Random Over-Sampling

- Randomly add more minority observations by replication

- No information loss

- Prone to overfitting due to copying same information