**Internship Number: DS2312**



**Glass Type Prediction**

Glass is a material that is widely used in several aspects of our everyday life and is very versatile. The primary elements are melted limestone, soda ash, and sand at high temperatures to create a liquid that can be molded and cooled into a solid. The first glass objects that have been discovered date to Mesopotamia and date back to approximately 3500 BCE. This technology has been developed over thousands of years.

Glass type also includes various natural elements and their oxides such as Sodium, Aluminum, Silicon and more. This article, we will explore the dataset, look into the exploratory data analysis part, various observations, feature engineering and finally we prepare the model.

**About Dataset:**

Criminology department often deal with a lot of glasses in their daily life. Identifying the glass type is one of the major tasks to go deeper into the investigation. Having a model that predicts the glass type helps a lot. This dataset speaks about the various samples of glasses with their chemical properties. After finalizing the dataset, it was credited to Vina Spiehler in 1987.

**Problem Definition:**

Forensic science believes that precisely determining the kind of glass discovered at crime scenes is essential to criminal investigations. Broken glass is frequently found as evidence in crimes including vandalism, hit-and-runs, and burglaries. Forensic experts may be able to connect these glass pieces to particular producers or suppliers by examining their chemical makeup, which would help with the inquiry. This blog uses a dataset that includes different chemical properties of glass samples to investigate the problem of glass type prediction for criminology purposes.

**Feature Information:**

The glass dataset contains chemical characteristics that depict the makeup of glass specimens. These characteristics are utilized for the identification and categorization of various glass types according to their intended purpose or qualities. The goal variable, or labels, classifies the glass into different categories such as Building Windows Float Processed, Building Windows Non-Float Processed, Vehicle Windows Float Processed, Vehicle Windows Non-Float Processed, Containers, Tableware, and Headlamps. This categorization is crucial for use in forensics and industry, as the composition of glass can give valuable information for criminal cases and product evaluations.

RI: Refractive index

Na: Sodium (weight percent in corresponding oxide, as %Na2O)

Mg: Magnesium (weight percent in corresponding oxide, as %MgO)

Al: Aluminum (weight percent in corresponding oxide, as %Al2O3)

Si: Silicon (weight percent in corresponding oxide, as %SiO2)

K: Potassium (weight percent in corresponding oxide, as %K2O)

Ca: Calcium (weight percent in corresponding oxide, as %CaO)

Ba: Barium (weight percent in corresponding oxide, as %BaO)

Fe: Iron (weight percent in corresponding oxide, as %Fe2O3)

Class Labels:

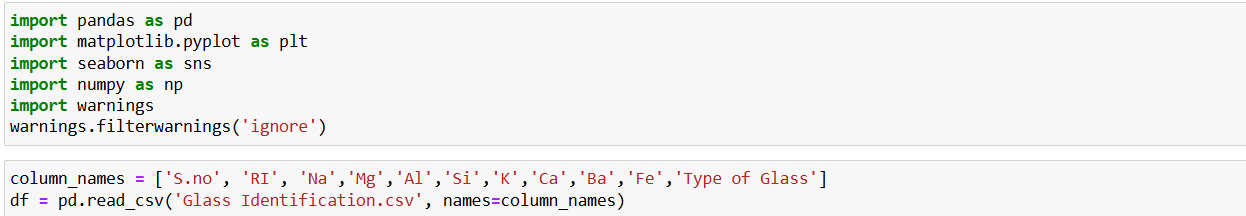
The target variable can classify the glass into one of several types:

* Building Windows Float Processed
* Building Windows Non-Float Processed
* Vehicle Windows Float Processed
* Vehicle Windows Non-Float Processed (NA in our dataset)
* Containers
* Tableware
* Headlamps

**Exploratory Data Analysis:**

Let’s start with some analysis using Python.

1. Extract the required libraries such as Pandas, Matplotlib, and Seaborn. Create a data frame for the dataset.



1. *Descriptive Statistics*:

The distribution of the dataset's form, central tendency, and dispersion is summarized by descriptive statistics. Below is a summary of the things we usually look for:

**Mean**: The mean represents the average value of each feature.

|  |
| --- |
|  |

* **Refractive Index (RI)**: The average refractive index is **1.51**
* **Sodium (Na)**: The average sodium content is **13.04**
* **Magnesium (Mg)**: The average magnesium content is **2.68**
* **Aluminum (Al)**: The average aluminum content is **1.44**
* **Silicon (Si)**: The average silicon content is **72.65**
* **Potassium (K)**: The average potassium content is **0.49**
* **Calcium (Ca)**: The average calcium content is **8.95**
* **Barium (Ba)**: The average barium content is **0.17**
* **Iron (Fe)**: The average iron content is **0.57**

.

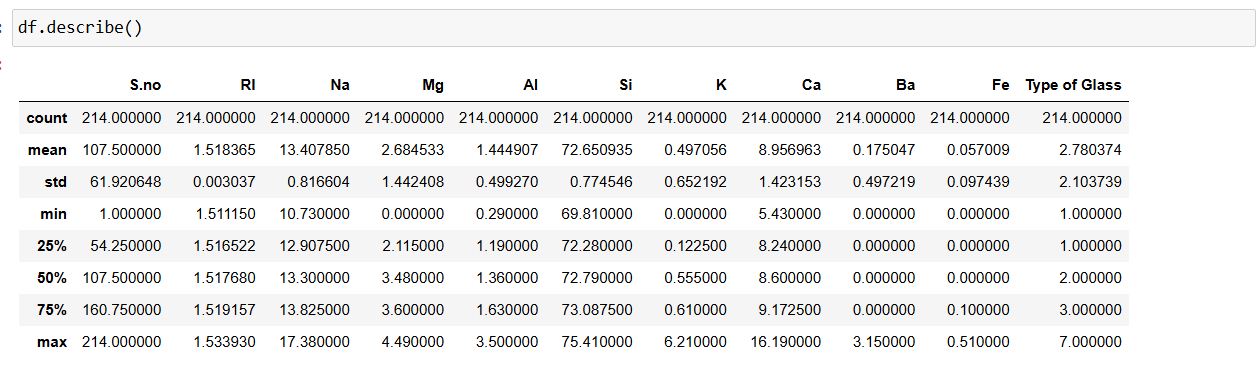
**Median**: Each feature's midpoint value.

**Standard Deviation**: A measurement of the feature's degree of fluctuation or dispersion.

* **Refractive Index (RI)**: The standard deviation is **0.003**.
* **Sodium (Na)**: The standard deviation is **0.82**.
* **Magnesium (Mg)**: The standard deviation is **1.44**.
* **Aluminum (Al)**: The standard deviation is **0.50**.
* **Silicon (Si)**: The standard deviation is **0.77**.
* **Potassium (K)**: The standard deviation is **0.37**.
* **Calcium (Ca)**: The standard deviation is **1.42**.
* **Barium (Ba)**: The standard deviation is **0.50**.
* **Iron (Fe)**: The standard deviation is **0.24**.
* **Type of Glass**: The standard deviation of glass types is approximately **2.10**.

**Min and Max**: Each feature's lowest and maximum values.

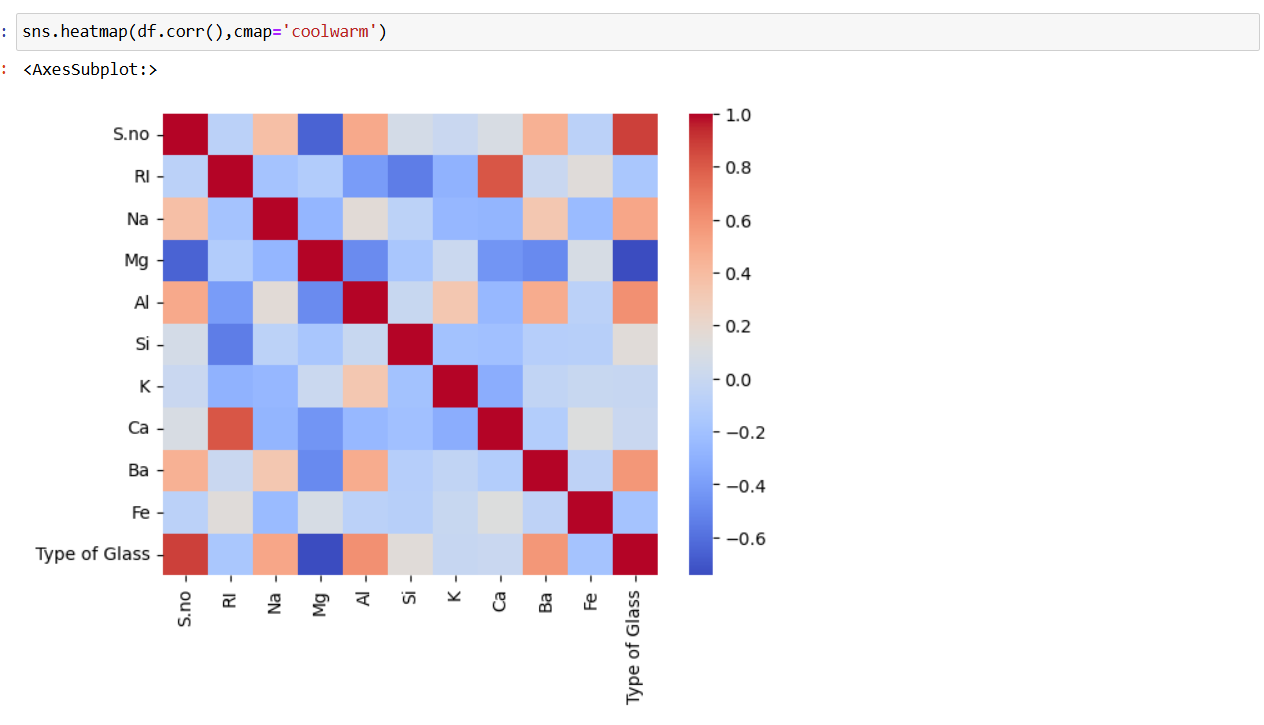
We can see any abnormalities and comprehend the overall pattern of the data with the aid of these statistics.



1. *Data Visualization*:

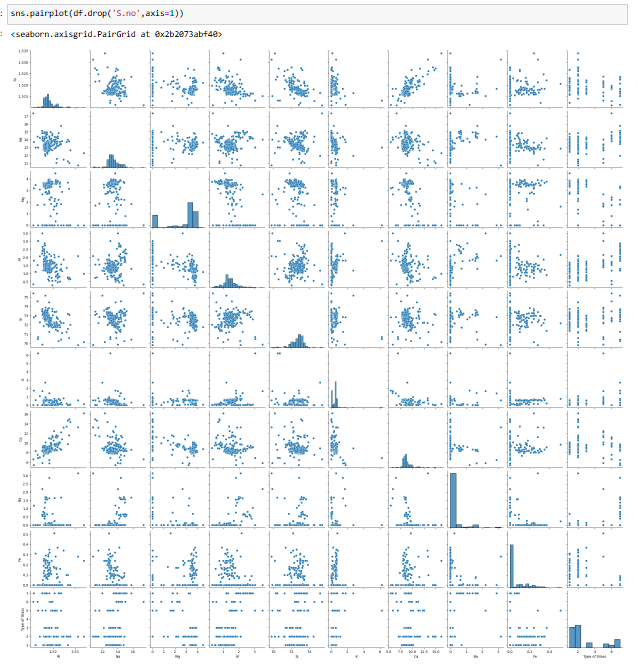
Visualizing the data in the form of graphs help us understand a more detailed about the data we have. These are the following graphs that give required data for our analysis,

* Heatmaps: Heatmaps are indispensable for gaining insights into the relationships among various features within a dataset. Within the glass prediction dataset, heatmaps illustrate the correlation matrix by showing the correlation coefficient between two features in each cell, which can range from -1 to 1. A heatmap is able to rapidly indicate characteristics with strong positive or negative correlations, which are essential for detecting multicollinearity and comprehending the connections between variables. On the other hand, features with little to zero correlation offer distinct information that could improve the model's efficiency. Heatmaps aid in feature selection and guide the feature engineering process by visualizing relationships to ensure the accuracy and interpretability of the predictive model.



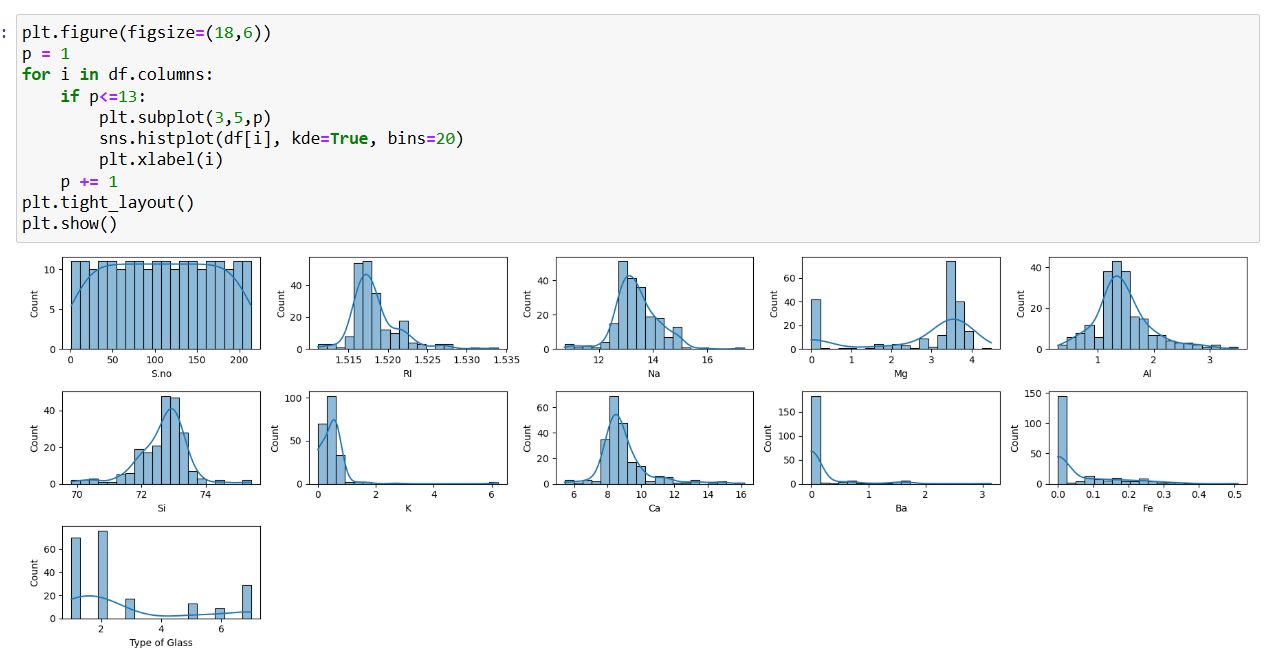
* Pair Plot: Pair plots are an effective way to visually show the connections between different pairs of features in a dataset all at once. Pair plots for the glass prediction dataset can unveil possible relationships and connections between the chemical characteristics such as Refractive Index, Sodium, Magnesium, Aluminum, Silicon, Potassium, Calcium, Barium, and Iron.

By graphing every combination of features, we can pinpoint groupings, patterns, and unusual data points that distinguish various glass categories. Clusters shown in the pair plot may suggest that specific combinations of features are common in certain glass types, while trends seen can indicate linear or non-linear connections. In general, pair plots offer a thorough examination of the dataset, aiding in improved feature selection and deepening the understanding of patterns present in the data.

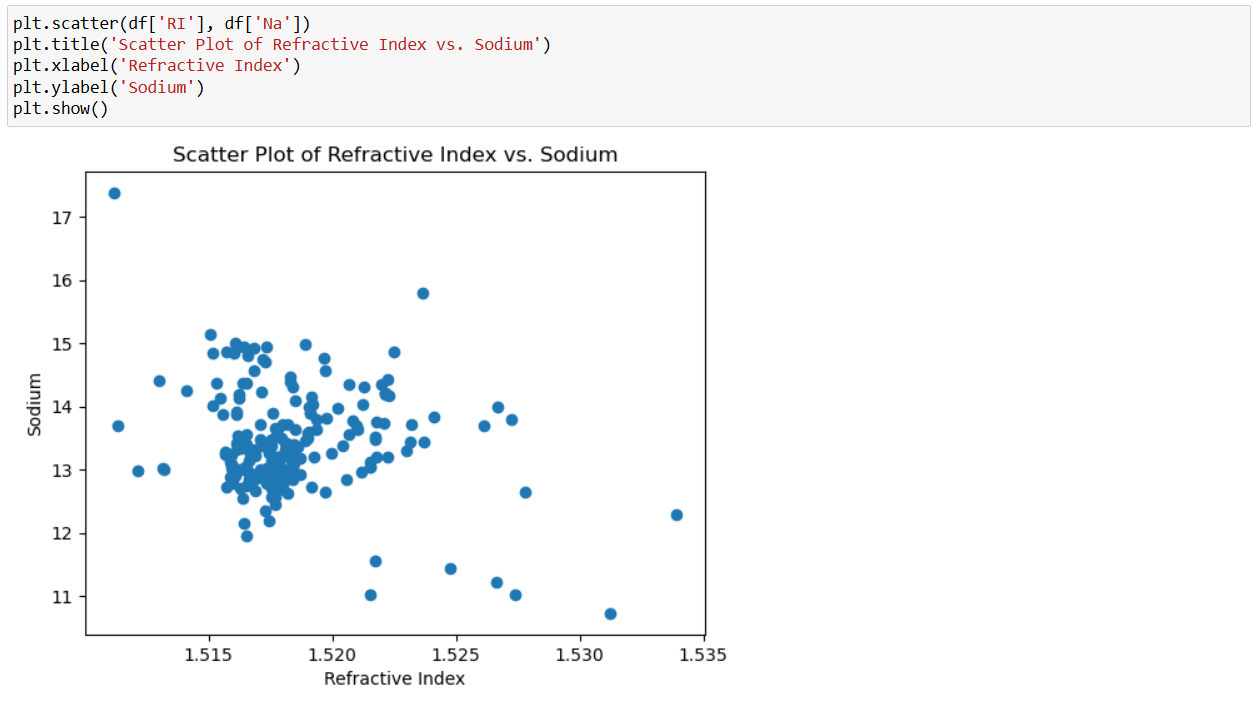


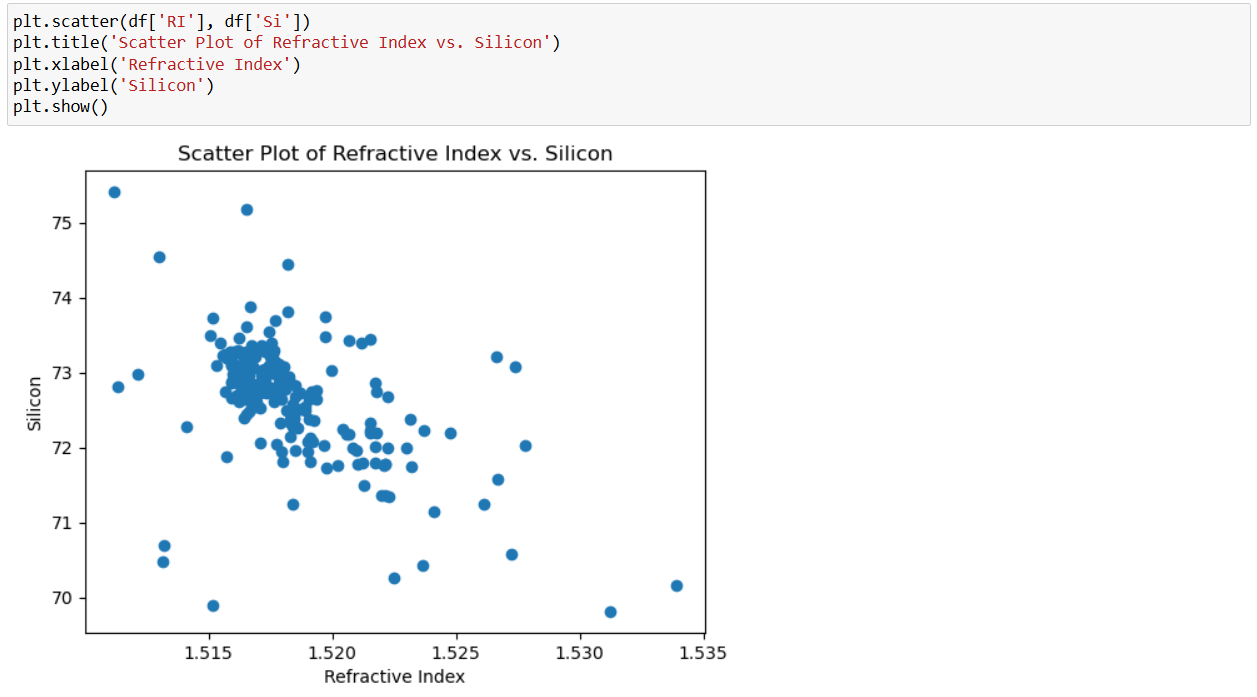
* Histograms: Histograms play a key role in exploratory data analysis by visually illustrating how a dataset's numerical characteristics are distributed. Histograms are useful in examining how chemical properties are distributed among various glass types within the glass prediction dataset.

Through analyzing these graphs, we can determine the central tendencies, variability, and identify any outliers present, all of which are essential for feature engineering and model building.



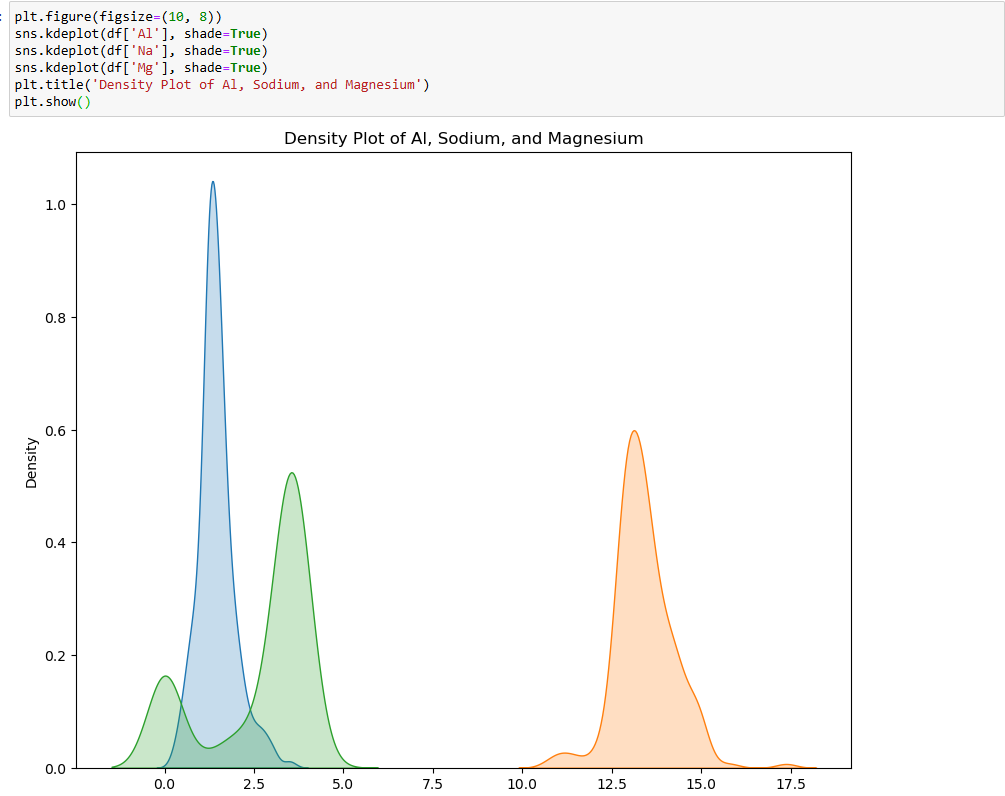
* Scatter Plots: Scatter plots serve as a useful visualization method for examining the correlation between two numerical characteristics. In the glass prediction dataset scenario, scatter plots can assist in comprehending the relationship and interplay among distinct chemical attributes in different glass varieties. Analyzing these graphs allows us to pinpoint patterns, trends, and possible outliers, which are essential for selecting features, engineering, and developing models.





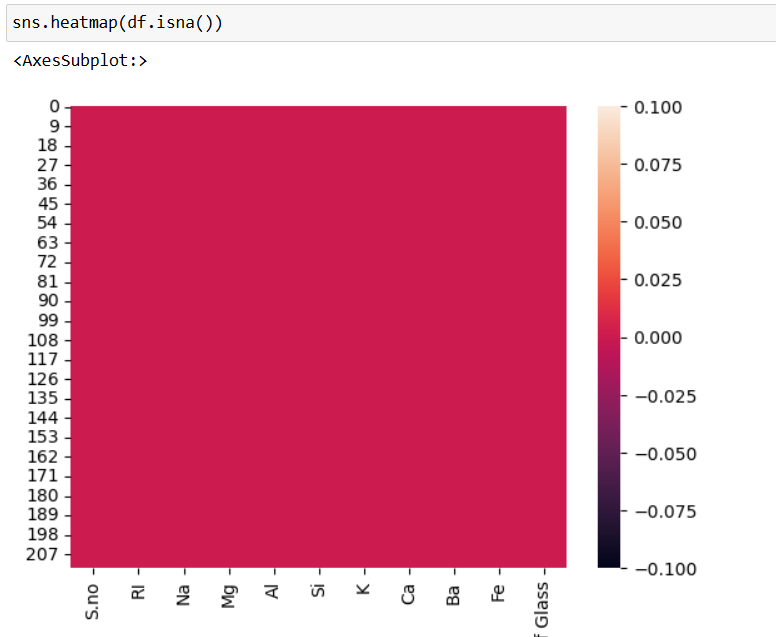
* Scatter Plots: Density plots, also referred to as kernel density estimation (KDE) plots, are a valuable tool in the process of exploring and analyzing data. They are employed to depict the spread of a continuous variable and grasp its fundamental probability density.

Within the glass prediction dataset, density plots are useful for understanding how chemical properties are spread out among the various types of glass. By analyzing these graphs, we can determine the distribution of features and whether there are any noticeable peaks or patterns that may assist in classification.

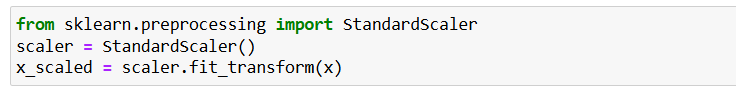


**Data Preprocessing:**

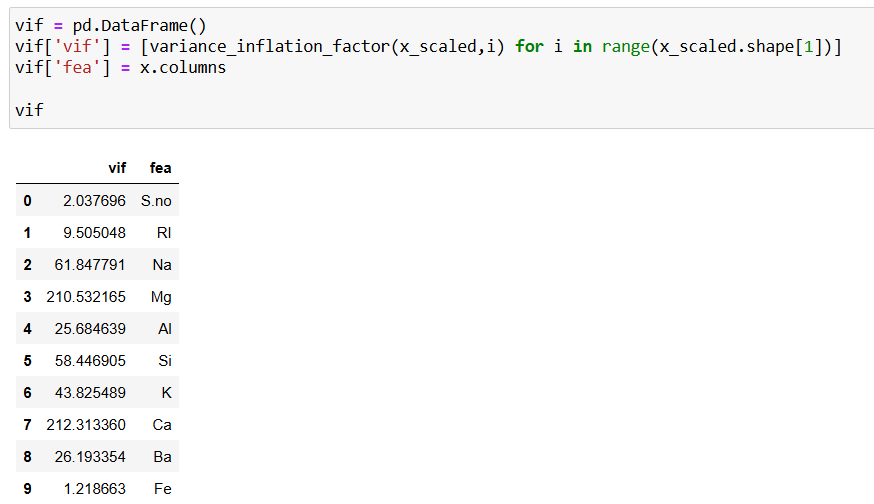
* In our dataset there are no nulls present, so no need to perform any operations related to dealing with nulls.



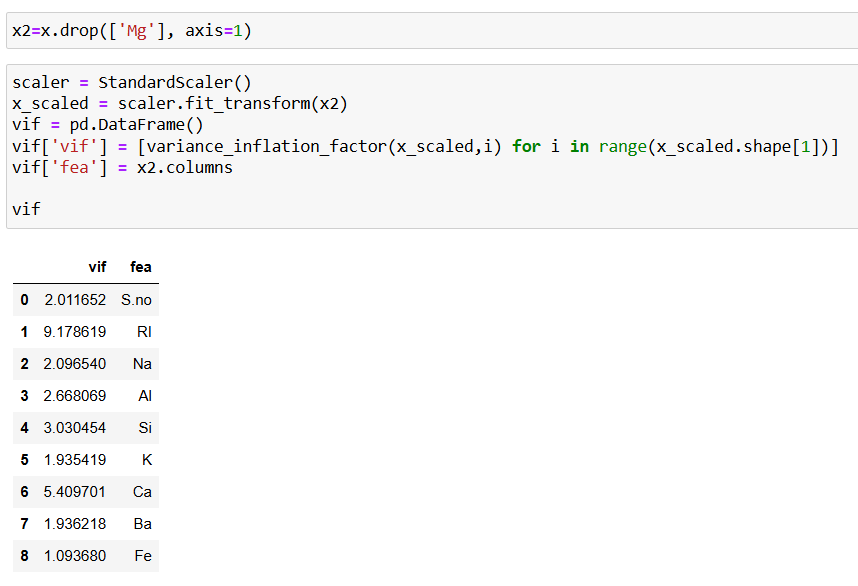
* Normalizing the data using standard scaler method in order to reduce the degree of magnitude.



* Let’s check the multicollinearity by finding out variance influence factor (VIF) of each feature.



* From observing the VIF of each feature, Magnesium has the max. VIF number. So, it is recommended to remove in order to mitigate multicollinearity.



* After removing the column, the VIF numbers seems to be in normal range as required.
* ***Encoding****:* The dataset has categorical columns, which need to be converted into numerical values. To do this we can use different Label encoders, but in this case we have only two categories so replace method is used to directly replace the selected categories.
* ***Oversampling:***

There is a total of seven label categories, out of which, one label category, ‘Vehicle Windows Non-Float Processed’ is not present in the dataset.

As there are more number of categories in the labels, we can reduce the classification by limiting it to two.

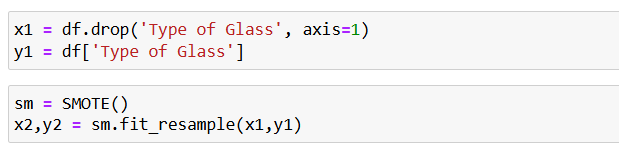
The classes, building windows float processed, building windows non float processed, and vehicle windows float processed can be combined into one category as window glass.

The remaining classes of glass can be categorized as non-window glasses.

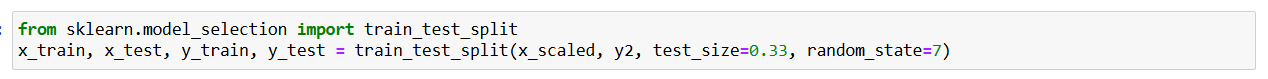
If we look into the labels, they are not symmetric, which needs to be made both classes of equal numbers.

This can be done using SMOTE method.

SMOTE resamples the dataset by oversampling the features and labels until the number of labels become symmetric.



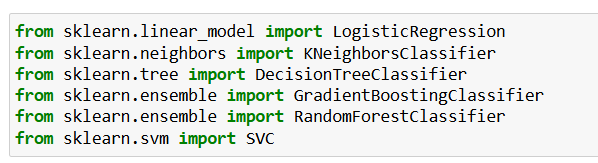
* ***Splitting the data***: Once oversampled and cleaned, the data needs to be divided into training set and test set. This is done by train\_test\_split method. Normally, training data is 65-80% of the overall data and the residue is for testing.

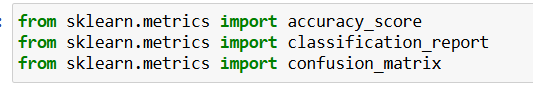


**Model Building:**

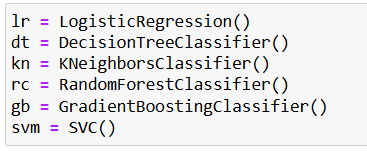
Now, our dataset is clear and have been spitted into training and testing datasets. It’s time to build a machine learning model to predict the type of glass.

* Importing all the required libraries and modules from sklearn and import all the evaluation metrics for the models.

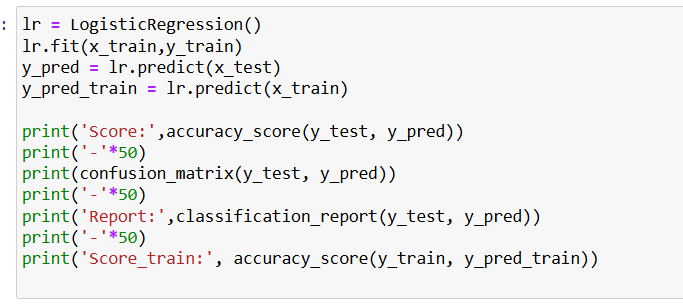




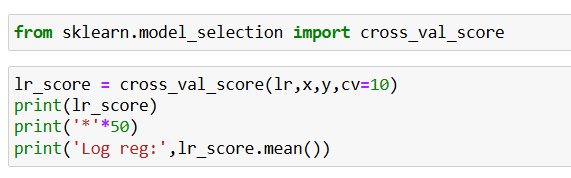
* Instantiate the models and start predicting by choosing different models.



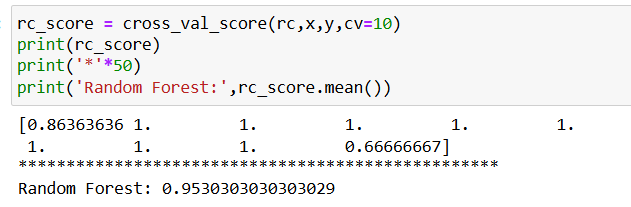
* Evaluate each model’s accuracy score and check the classification report with confusion matrix.



* After this, we need to select the best model, to do this let’s perform cross validation by importing necessary modules



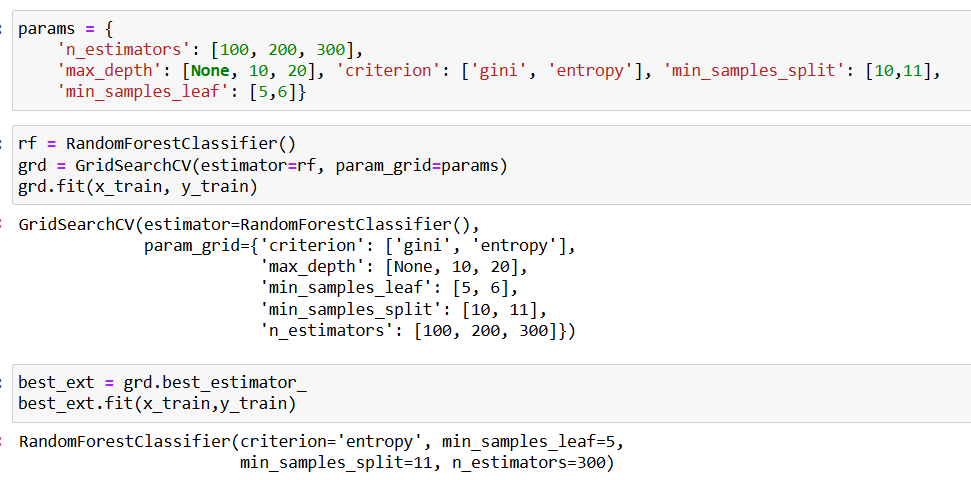
* By observing the accuracy scores, Random Forest classification method gives better accuracy compared to others which is 0.953. Hence, we are finalizing this as the best model.



**Optimize the Model:**

As the best model we selected is Random Forest, it could be even made better by optimizing it using hyper parameter tuning.

* Importing GridSearchCV from sklearn’s model selection
* Once imported, create a parameter grid and add the required hyper parameters
* After searching the best parameters, fit the same into the best model



**Save the Model:**

It is crucial to upkeep a well-trained machine learning model in order to use it in practical situations and generate predictions whenever needed. Utilizing the pickle module, which stores and retrieves Python objects in memory, is a widely used method for saving a model in Python because of its ease and efficiency.

Once you have finished training a model, you have the option to save it to a file using pickle.dump(), which involves converting the entire object into a byte stream for serialization. This serialized model can be conveniently saved and retrieved for fast deployment without the need for retraining. Utilizing pickle.load() to reload the model into memory gives you the ability to make predictions on fresh data.

Pickle's major benefit is its user-friendliness as it can handle a variety of Python objects and intricate data structures effortlessly. This feature is especially beneficial for storing trained machine learning models, which frequently contain complex settings and extensive arrays of parameters.



**Conclusion:**

The blog post discussed predicting glass types by examining their chemical properties, crucial in forensic cases involving broken glass. The analysis began by studying a dataset containing characteristics like refractive index and oxide levels of elements such as Sodium, Magnesium, and Aluminum. Exploratory data analysis (EDA) provided valuable insights, with histograms displaying normal distributions for Sodium, Aluminum, and Silicon. Pair plots suggested connections between features like Calcium and Barium, indicating similar influences on glass types.

The correlation matrix highlighted strong relationships, like the positive correlation between Calcium and Barium. Machine learning models, including Logistic Regression and Random Forest, were constructed and evaluated, with Random Forest achieving 95.3% accuracy. The study demonstrated the efficacy of Python's sklearn library for predictive modeling in forensic science, underscoring the importance of EDA in developing accurate glass type classification models for potential real-time forecasting.