**DS USING PYTHON LAB**

**EXPERIMENT: 02**

**AIM:**

To perform following data visualization and exploration on your selected dataset:

1. Create bar graph, contingency table using any 2 features.

2. Plot Scatter plot, box plot, Heatmap using seaborn.

3. Create histogram and normalized Histogram.

4. Describe what this graph and table indicates.

5. Handle outlier using box plot and Interquartile range

**THEORY:**

**Selected Dataset :**<https://www.kaggle.com/datasets/spscientist/students-performance-in-exams?resource=download>

The above data set consists of the marks secured by the students in various subjects.From this data set we can understand the influence of the parents background, test preparation etc on students performance

**Seaborn:**

Seaborn is an amazing visualization library for statistical graphics plotting in Python. It provides beautiful default styles and color palettes to make statistical plots more attractive. It is built on the top of matplotlib library and also closely integrated to the data structures from pandas.

Seaborn aims to make visualization the central part of exploring and understanding data. It provides dataset-oriented APIs, so that we can switch between different visual representations for the same variables for better understanding of the dataset.

Seaborn divides plot into the below categories –

* Relational plots: This plot is used to understand the relation between two variables.
* Categorical plots: This plot deals with categorical variables and how they can be visualized.
* Distribution plots: This plot is used for examining univariate and bivariate distributions
* Regression plots: The regression plots in seaborn are primarily intended to add a visual guide that helps to emphasize patterns in a dataset during exploratory data analyses.
* Matrix plots: A matrix plot is an array of scatterplots.
* Multi-plot grids: It is a useful approach to draw multiple instances of the same plot on different subsets of the dataset.

Import the Seaborn module in your code using the following statement:

import seaborn as sns

**IMPLEMENTATION:**

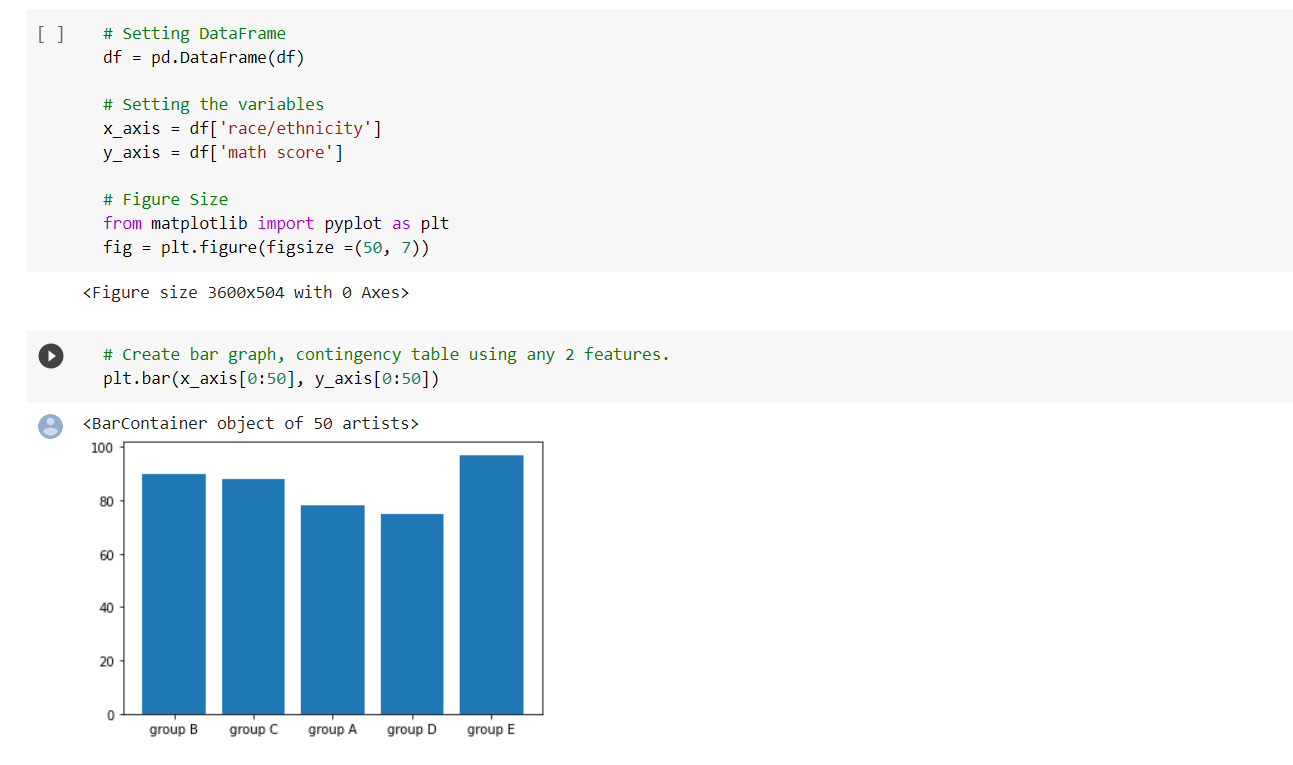
1. **Bar graph**

A bar plot or bar chart is a graph that represents the category of data with rectangular bars with lengths and heights that is proportional to the values which they represent. A bar chart describes the comparisons between the discrete categories.

The matplotlib API in Python provides the bar() function which can be used in MATLAB style use or as an object-oriented API.

Syntax:

plt.bar(x, height, width, bottom, align)



1. **Contingency table using any 2 features**

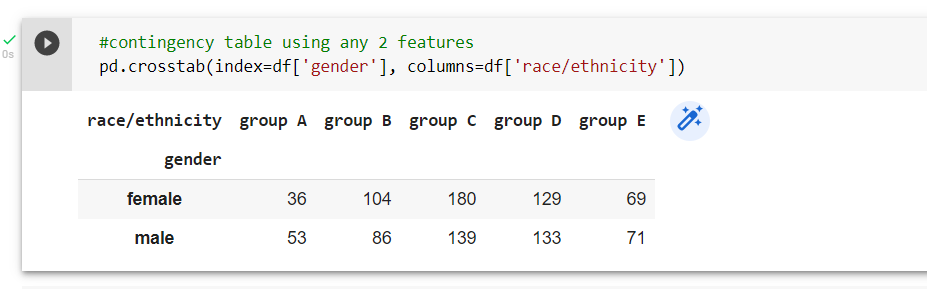
A contingency table is a type of table that summarizes the relationship between two categorical variables.To create a contingency table in Python, we can use the pandas.crosstab() function, which uses the following syntax:

pandas.crosstab(index, columns)

where:

index: name of variable to display in the rows of the contingency table

columns: name of variable to display in the columns of the contingency table



1. **Scatter plot using seaborn**

Scatterplot can be used with several semantic groupings which can help to understand well in a graph. They can plot two-dimensional graphics that can be enhanced by mapping up to three additional variables while using the semantics of hue, size, and style parameters.

Syntax:

df = pd.read\_csv(‘data\_set.csv’)

sns.scatterplot(data = df, x = " ", y = " ")

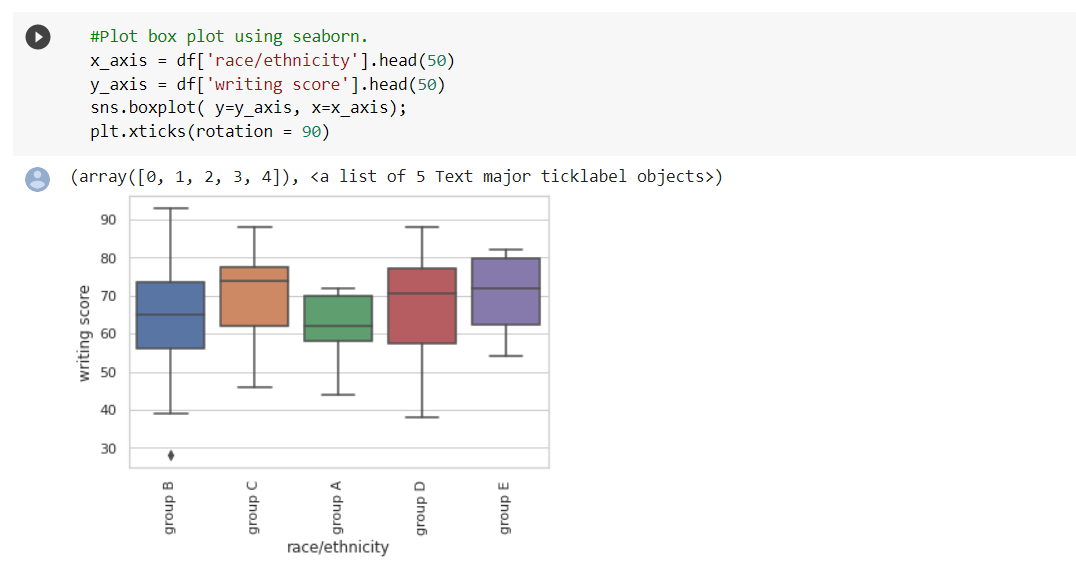


1. **Box plot using seaborn**

Boxplots are a great statistical tool for visualizing data and are commonly used during the Exploratory Data Analysis (EDA) phase of data science projects. They provide us with a quick statistical summary of the data, help us understand how data is distributed and help identify anomalous data points (outliers).

Syntax:

sns.boxplot(x=df[' '], y=df[' ']);

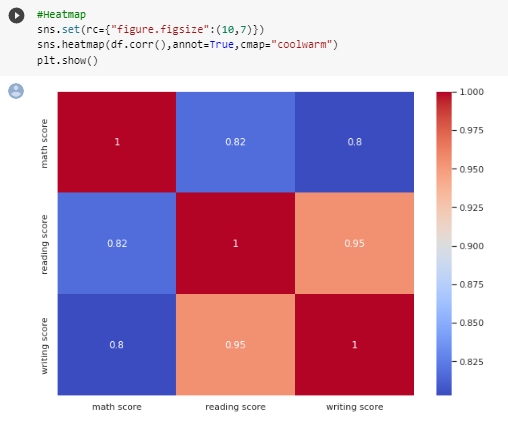


1. **Heatmap using seaborn.**

Heatmap is defined as a graphical representation of data using colors to visualize the value of the matrix. In this, to represent more common values or higher activities brighter colors basically reddish colors are used and to represent less common or activity values, darker colors are preferred. Heatmap is also defined by the name of the shading matrix.

Syntax:

sns.heatmap()



1. **Histogram and normalized Histogram.**

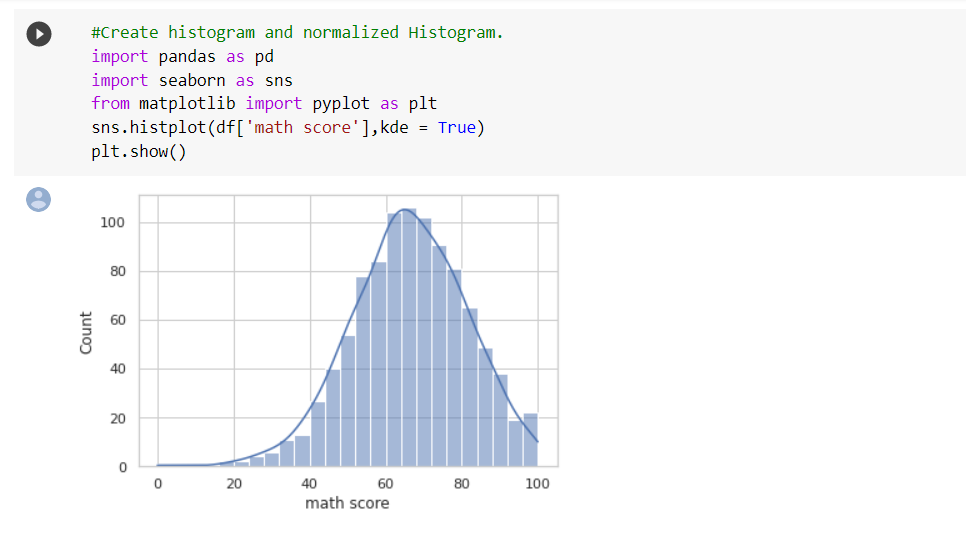
A histogram is used to analyze the probability distribution of univariate numerical data by plotting the count of the data instead of the values.

Histogram divides the entire range of values into a series of intervals called bins.

It then counts the number of values that fall in each bin and visualizes the results intuitively.

Syntax:

sns.histplot(data=df, x=df[“ "])

****

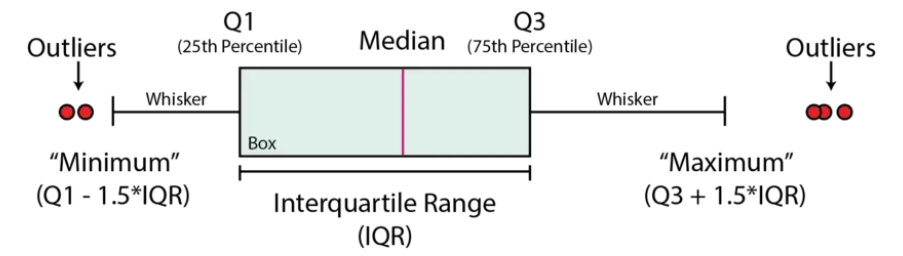
1. **Handle outlier using box plot and Interquartile range**

A boxplot is a graphical and standardized way to display the distribution of data based on five key numbers:

* minimum
* 1st Quartile (25th percentile)
* median (2nd Quartile/ 50th Percentile)
* 3rd Quartile (75th percentile)
* maximum

The minimum and maximum values are defined as Q1–1.5 \* IQR and Q3 + 1.5 \* IQR respectively. Any points that fall outside of these limits are referred to as outliers.

Graphical depiction of a boxplot highlighting key components, including the median, quartiles, outliers, and Interquartile Range.

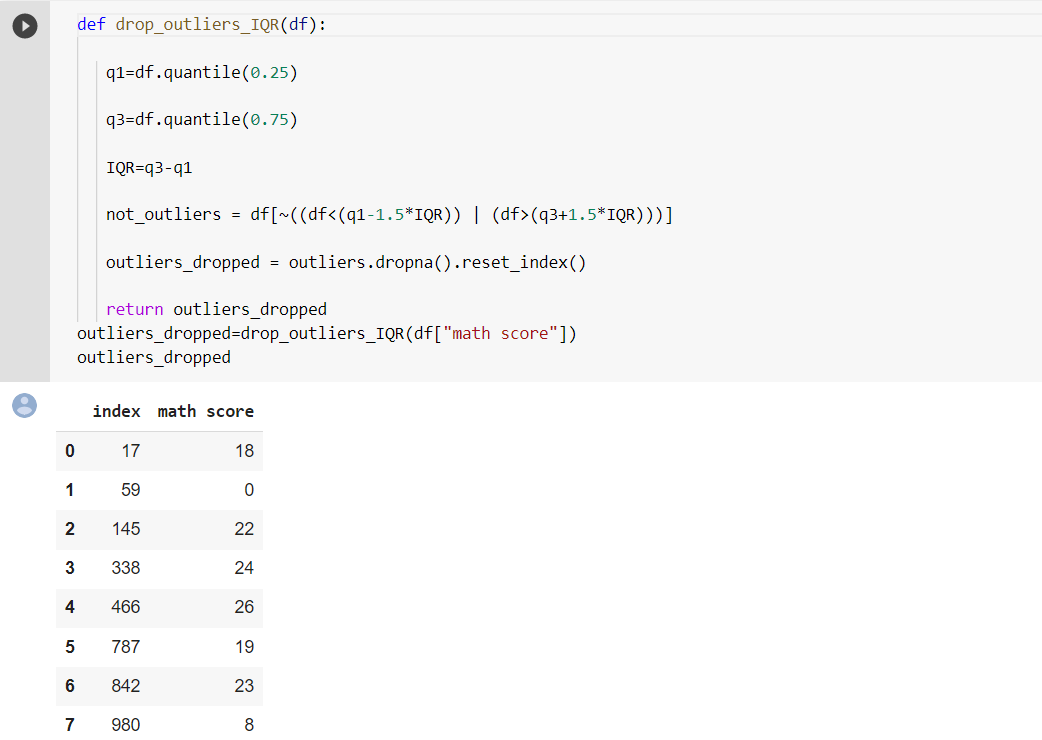


We calculated the outlier data points using the statistical method called interquartile range (IQR) instead of using Z-score. Using the IQR, the outlier data points are the ones falling below Q1–1.5 IQR or above Q3 + 1.5 IQR. The Q1 is the 25th percentile and Q3 is the 75th percentile of the dataset, and IQR represents the interquartile range calculated by Q3 minus Q1 (Q3–Q1).

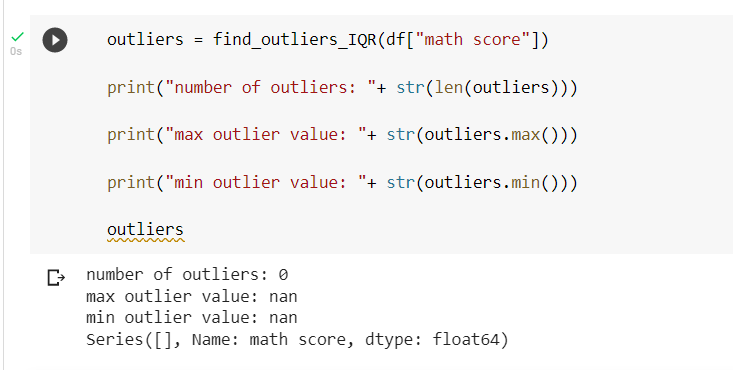
Using pandas .quantile() function, we created simple Python function that takes in our column from the dataframe and outputs the outliers:

****

Using this method, we essentially drop all the outliers from the data, excluding them from the analysis and modeling. Inside the function we create a dataframe named not\_outliers that replaces the outlier values with a NULL. Then we can use .dropna(), to drop the rows with NULL values.

****

After dropping the outliers, we reexamine the statistics.

****

**CONCLUSION:**

In this experiment,we studied the seaborn library and we performed data visualization and exploration of our dataset using seaborn.