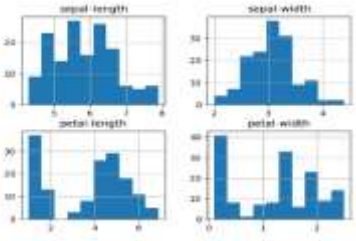
Tebibu Kebede

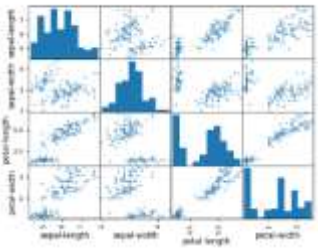
1.



2. Total Size of the Dataset**: The dataset size in terms of a matrix can be determined by the number of rows (instances) and columns (attributes). The Iris dataset has 150 instances and 5 attributes, so its size is 150×5**

3. Command to Print First 10 Rows**: To see the first 10 rows of the dataset, you can use the command dataset.head(10) in Python**

4.



5. 3.054000

6. 2.500000

7. Support Vector Machines (SVM)

8. Number of Classes in the Dataset**: The Iris dataset has 3 classes: Iris-setosa, Iris-versicolor, and Iris-virginica.**

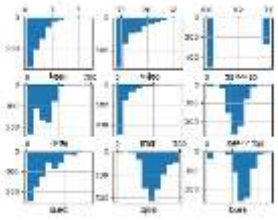
9. Univariate plots show data distribution of a single variable. For example, histograms and box plots in the Iris dataset project show the distribution of attributes like sepal length, sepal width, petal length, and petal width individually.

Bivariate plots (or multivariate) show the relationship between two (or more) variables. The scatter plot matrix in this project, for example, shows how each pair of attributes

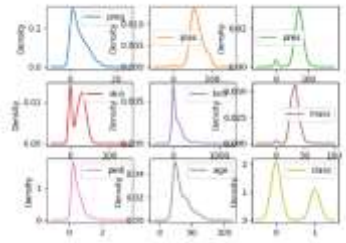
interact with each other, allowing for the observation of correlation patterns, trends, and clusters.

10. Training and Test Dataset Distribution: In the project, the dataset is split into a training set and a test (or validation) set. The training set consists of 80% of the data, used for training the models. The remaining 20% serves as the test set, used for validating the model's performance. This split is achieved with the command train_test_split(X, y, test_size=0.20, random_state=1).

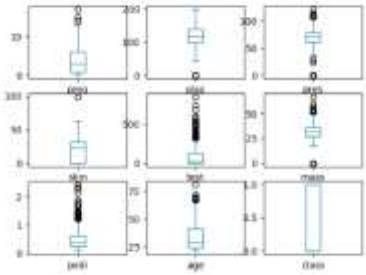
11. a.



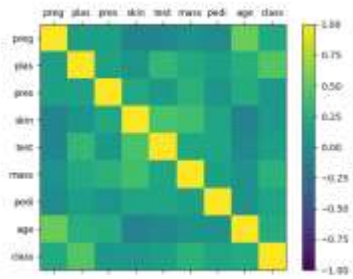
b.



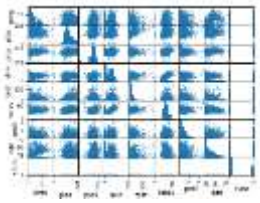
c.



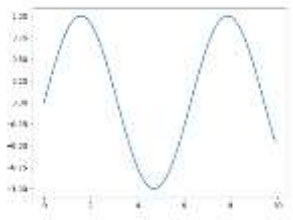
D.



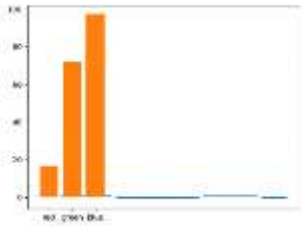
e.



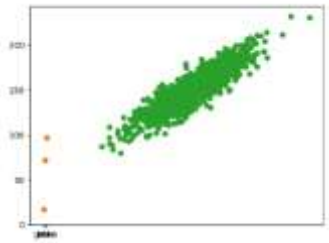
13.



14.



15.



16. There are two classes present in the dataset.

17. To change the ratio of training and test dataset distribution, you can use the `train_test_split` function from scikit-learn and specify the `test_size` parameter to control the ratio. Here's an example:

```
from sklearn.model_selection import train_test_split # Split the
dataset with a 70% training and 30% test ratio X_train, X_test,
y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=1)
```

In this example, `test_size=0.3` means that 30% of the data will be used for testing, and the remaining 70% will be used for training. You can adjust the `test_size` parameter to change the ratio as needed.

18. `KNN: 0.957191 (0.043263) and SVM: 0.983974 (0.032083)`

19. The difference between Density Plots and Histograms:

- Density Plots: Density plots are smooth representations of the distribution of data and are often created using kernel density estimation. They provide a continuous view of data density and are helpful for visualizing the underlying distribution.
- Histograms: Histograms divide data into discrete bins and count the number of data points in each bin. They represent data in a binned and discrete form, providing a clearer view of data frequencies.

20. Three advantages of Box-Whisker Plots and Correlation Matrix plots:

- Box-Whisker Plots:
    1. Outlier Detection: Box plots can easily identify outliers in data by displaying them as individual points beyond the whiskers.
    2. Visualizing Spread: They provide a clear representation of the spread and variability of data, including quartiles and median.
    3. Comparison: Box plots allow for easy comparison of multiple datasets or groups, making it useful for comparative analysis.
- Correlation Matrix Plots:
    1. Visualizing Relationships: Correlation matrix plots help visualize relationships between variables, showing the strength and direction of correlations.
    2. Identifying Multicollinearity: They are useful for identifying multicollinearity (high correlations between independent variables) in regression analysis.
    3. Data Selection: Correlation plots can guide variable selection by highlighting which variables are most correlated with the target variable.