Source Code Explanation :

1. Classification.py

Class name : Classification_descisionTree

__init__ :

1. Loads the dataset.
2. Drops the unnecessary columns.
3. Separates the Class label and drops it from the main data set.

Functions :

1. Question1_i : Calculates number of instances and returns it.

2. Question1_ii : Calculates null values and returns it.

3. Question1_iii : Calculates Fraction of missing values and returns it.

4. Question1_iv : Calculates number of instances with missing values and returns it.

5. Question1_v : Calculates Fraction of instances with missing values over all instances and returns It.

6. descritization : Performs label Encoding on the class labels and the data set and returns it.

7. Question2 : Returns Descritized class labels with Label Encoder.

8. main : Main Function that prints all the results on the console.

9. __descritization : Performs label encoding on the data set and returns an Encoded data set.

10. dataPreProcessing_Q_3 : Process the data set and class label for train test split and returns new data set and class label data as X and y

11. dataSplitting : Splits X and y to x train y train x test y test as well as label splitting to y train label and y test label and returns the data.

12. DtreeClassifier : Declares and initialises the descision tree classifier.

13. predict : Takes Descision tree as an argument and makes predictions.

14. ClassificationReport : Returns classification report based on test data predictions.

15. Confusion_Matrix_Error_Rate : Calculates the confusion matrix and error rate and returns the values.

16. D_prime : Creates a new dataset based on the original dataset and returns it.

17. D_one_prime : Creates D_1_prime based on D_prime and the properties defined in the assignment specification.

18. D_two_prime : Creates D_2_prime based on D_prime and the properties defined in the assignment specification.

19. D_prime_data_splitting : Splits D_prime to create new training and testing samples for d1 prime and d2 prime classifiers.

20. D_prime_data_preprocessing_splitting: Performs preprocessing of data like label and dataset encoding on the new d1 prime and d2 prime test and train samples as well the associated class labels.

21. __D_prime_descritization : Performs label encoding on the data samples for d1prime and d2prime datasets.

22. Instructions to run the program : On the CLI run : python3 Classification.py

23. Dependencies :

1. pandas, matplotlib , sklearn , pprint libraries.
2. python3 environment

2. Clustering.py

Class name : Classification_descisionTree

__init__ :

1. Loads the dataset.
2. Drops the unnecessary columns.
3. Separates the Class label and drops it from the main data set.

Function :

1. question1 : Calculates mean , min and max for each attributes and returns it.

2. question2 :
1. Makes K mean classifier with k = 3
2. Stores the classifier , labels , cluster centers and inertia (SSE) to variables.
3. Loops through the data and plots scatter plot for each pair of attributes.

3. KmeanClassifier : Declares and initialises the k means classifier , calculates the labels , cluster centers , intertia (SSD) and distSpace . Then it returns those values.

4. plot_data : Declares new variables and assigns each attribute values I.e Values of each column in the data set to be used for creating the scatter plot.

5. Scatter_Plot : Takes figure number, data , label , K mean classifier , xlabel , ylabel and cluster centers as arguments to create individual scatter plots.

6. Question3 : Creates k means classifier with k in the set of (3,5,10) . Calculates WC , BC and Calinski index and returns the values as well as plots the heatmap table.

7. k_means_algorithm_loop : Creates K means classifier in a loop and returns an object containing necessary values like the classifiers , cluster centers , labels .

8. get_CH : Calinski – Harabasz score for the classfiers and returns the values. Takes the data and labels as arguments.

9. get_BC : Calculates the between cluster distance for each classfier and returns the value.

10. main : Prints all the necessary values to the console and creates all the visualisations and graphs.

11. Instructions to run the program : On the CLI run : python3 Clustering.py

12. Dependencies :

1. pandas, matplotlib , sklearn , numpy libraries.
2. python3 environment