Self-Reflection

Student ID: 123456

Contributions to the Team Project:

Throughout the team project, I actively contributed to various aspects, including preprocessing the dataset, training machine learning models, exploring unsupervised learning techniques, and experimenting with neural networks. Here's a breakdown of my contributions:

1. Preprocessing:

- I took the initiative to devise a plan for preprocessing the dataset. Before diving into implementation, I emphasized visualizing the data to understand its structure and characteristics thoroughly.

- Ensured that the data was organized into a DataFrame, ready for model training.

Did following in our code:

- Data Cleaning: Implemented robust procedures for handling missing values, ensuring that our dataset was free from any inconsistencies that could potentially affect model performance.
- Label Encoding: Employed label encoding to convert categorical class labels into numerical values, a crucial step for compatibility with machine learning algorithms.
- Feature Scaling: Recognizing the significance of feature scaling in optimizing model performance, I utilized robust scaling techniques to normalize the range of features and mitigate the impact of outliers.
- Outlier Removal: Implemented outlier detection and removal techniques to enhance the robustness of our models, ensuring that they were less susceptible to the influence of anomalous data points.
- Feature Engineering: Leveraging domain knowledge, I combined relevant features to create new composite features, enriching our dataset with additional information while reducing dimensionality.

2. Model Development:

- Conducted a comprehensive study of various machine learning models before their implementation. This involved understanding the strengths and weaknesses of each model and assessing their suitability for our project.

- Actively participated in training and evaluating different models. Notably, KNN and logistic regression emerged as top performers, achieving accuracy rates of over 90%.

- Recognized the importance of selecting the right model for real-world scenarios, considering factors such as scalability and computational efficiency.

- Model Comparison: Collaborated with team members to compare the performance of different models, identifying logistic regression and KNN as standout performers with accuracy rates exceeding 90%.

3. Unsupervised Learning:

- Explored unsupervised learning techniques, including K-means clustering and hierarchical clustering.

- Applied K-means clustering, determining the number of clusters based on prior knowledge. Evaluated clustering quality using silhouette scores and gained insights into the separation of clusters within the data.

- Utilized hierarchical clustering and visualized dendrograms to understand the hierarchical structure of the data.

- Computed silhouette scores for individual features to assess their relevance in clustering.

4. Feature Relevance Analysis:

- Conducted an in-depth analysis to determine the relevance of features in the dataset.

- Identified features with high relevance, such as 'CO/SF3' and combined features, through silhouette scores.

- Recognized the importance of feature relevance in understanding the underlying structure of the data.

5. Neural Networks:

- Explored neural network models to broaden our understanding of deep learning concepts.

- Developed a sequential model using TensorFlow for learning purposes, although its performance on the dataset was comparable to other machine learning models.

- Gained insights into hyperparameter tuning and experimented with TensorFlow features like Tensor Board.

Key Learnings:

- Emphasized the critical role of preprocessing in model performance, highlighting the need to understand the data thoroughly before training models.

- Recognized the importance of selecting the right machine learning model for real-world applications, considering factors beyond just accuracy.

- Explored various unsupervised learning techniques and gained insights into clustering and feature relevance analysis.

- Broadened knowledge in deep learning concepts through experimentation with neural networks and TensorFlow.

←-- end --→

These are some of the Notes I took while developing code:

Preprocessing:

- Preparing plan how to implement preprocessing but visulizations of data first, describing the data and checking data at every point of time.
- Preprocessing steps involves:
  - Dropping na values
  - Encoding lables
  - Scaling featues
  - Removing outliers
  - Combining features(without using PCA)
  - Followed by organizing data into dataframe

In preprocessing I learned before training model its very critical to preprocess the data, also we need to check for different scalers that best fit the data to scale it, and importance of understanding the data beforehand.

Note – also learned mounting drive into colab which is not that big thing but it helps in running colab everytime without adding dataset again and again into cache memory.

Training the model:

- After preprocessing its very import to select the model so studying each model before implementation is very important
- In our case KNN and logistic regression provides best accuracy 90+
- But in real-world scenario we don't have that luxury to train each and every model we need to know which model we are training and what we can expect, because realworld data will be very much bigger.
- Also, I haven't used ISLP package due to 2 reasons:
    - It taking too much time to install now a days
    - Also, ISLP is developed for teaching purposes.
- We can able to check Classification report to check for f1 score primary as it gives overall health of model. If f1 score is less accuracy might not matter as model is working very badly.

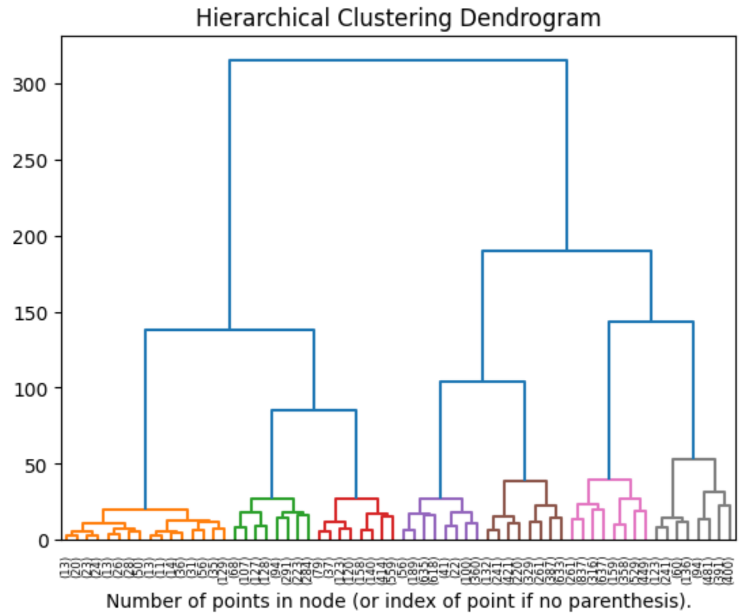
Unsupervised Learning:

- For kmeans we already know no of clusters to form so we just applied same as n_clusters, guessing other number will not provide good results.
- Computed silhouette score that provides how clusters are separated into the data it ranges from -1 to 1 where values closer to 1 is good sign.
- Applied Hierarchical clustering as well where by plotting dendogram we can now see no of features in the data so plotted that as well
- Computed individual silhouette score to check for feature relevance.

Results for feature relevance:

above answers - [('MinorAxisLength', 0.5180661128054215), ('Extent', 0.5039373999538688), ('Solidity', 0.5229329116305274), ('roundness', 0.4787921703178318), ('ShapeFactor1', 0.5091863595985126), ('ShapeFactor2', 0.4768499268189141), ('ShapeFactor4', 0.5146187175630692), ('AR/ECC', 0.5179177453611307), ('AR/PE/MA/ED/CA', 0.5193765976839253), ('CO/SF3', 0.5159676721970078), ('Class', 1.0)]

we can see 'CO/SF3' has highest relevance followed by combined features



Clusters are well separated if we cut around 55

Also plotted covariance matrix it gives inappropriate values as we preprocessed data manually if we did PCA it might be better.

Neural networks:

Just for the sake of learning I also developed sequential model for neural nets, although for this dataset it does not makes much sense as accuracy is same as other machine learning models

But from that I learned hyperparameters turning and some deep learning concepts like how neural networks works how to use tensorflow board etc..