Machine Learning
Project Document

Student Name	Srikanth G
Batch	AI Elite 17
Project Name	Dry Been Classification
Project Domain	Consumer Products
Type of Machine Learning	Supervised
Type of Problem	Classification/Regression
Project Methodology	MLDL-C
Stages Involved	Problem statement,,Data collection, EDA, preprocessing, EDA, Training, Evaluation.

Problem Statement:

Business Understanding: Problem statement: Classification of Dry beans. Thanks to their high nutritional value and ability to be used in a variety of culinary preparations, dry beans are a staple meal that is widely consumed worldwide. However, because dry beans vary in size, colour, and appearance, it can be difficult to ensure their quality and classification.

The main goal of the problem statement is to create a reliable system for classifying dry beans according to different characteristics. With the help of this categorization method, various kinds of dry beans should be precisely categorised, offering useful information to distributors, customers, and farmers alike.

Business constraints: Accuracy Requirements: To identify various types of dry beans, the classification model needs to meet quality standards, which include a certain level of accuracy. Maintaining product quality and obtaining trustworthy classification results require doing this

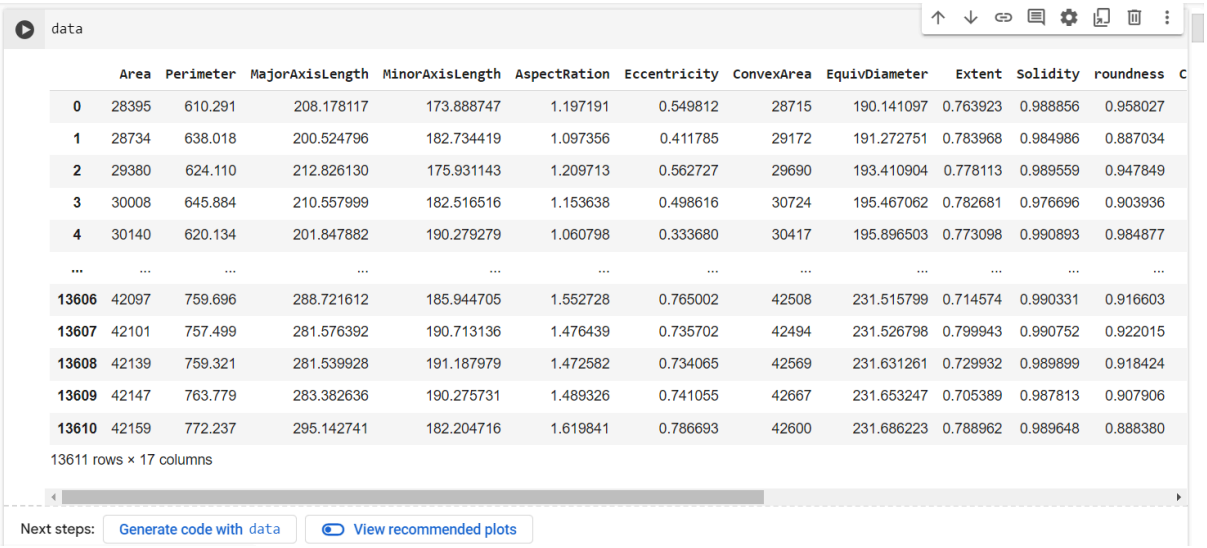
Stage 1: Data Collection and Understanding

a) Data Collection: Data has been collected from Kaggle website.

https://www.kaggle.com/datasets/muratkokludataset/dry-bean-dataset

Apart from this there are other means of Data collection like API, Web scraping or Manual collection of data.

b) Data Understanding: Dry bean dataset has 13611 data points with 16 feature variables and one class variable. The features are giving the specifications of the bean based on which we are going to classify the class of bean. The various class labels of bean provided in the data set are, Dermason, Sira, Seker, Horoz, Cali, Barbunya, Bombay.



data

	Area	Perimeter	MajorAxisLength	MinorAxisLength	AspectRation	Eccentricity	ConvexArea	EquivDiameter	Extent	Solidity	roundness	C
0	28395	610.291	208.178117	173.888747	1.197191	0.549812	28715	190.141097	0.763923	0.988856	0.958027	
1	28734	638.018	200.524796	182.734419	1.097356	0.411785	29172	191.272751	0.783968	0.984986	0.887034	
2	29380	624.110	212.826130	175.931143	1.209713	0.562727	29690	193.410904	0.778113	0.989559	0.947849	
3	30008	645.884	210.557999	182.516516	1.153638	0.498616	30724	195.467062	0.782681	0.976696	0.903936	
4	30140	620.134	201.847882	190.279279	1.060798	0.333680	30417	195.896503	0.773098	0.990893	0.984877	
...	...	...	...	...	...	...	...	...	...	...	...	
13606	42097	759.696	288.721612	185.944705	1.552728	0.765002	42508	231.515799	0.714574	0.990331	0.916603	
13607	42101	757.499	281.576392	190.713136	1.476439	0.735702	42494	231.526798	0.799943	0.990752	0.922015	
13608	42139	759.321	281.539928	191.187979	1.472582	0.734065	42569	231.631261	0.729932	0.989899	0.918424	
13609	42147	763.779	283.382636	190.275731	1.489326	0.741055	42667	231.653247	0.705389	0.987813	0.907906	
13610	42159	772.237	295.142741	182.204716	1.619841	0.786693	42600	231.686223	0.788962	0.989648	0.888380	

13611 rows × 17 columns

Next steps: Generate code with data View recommended plots

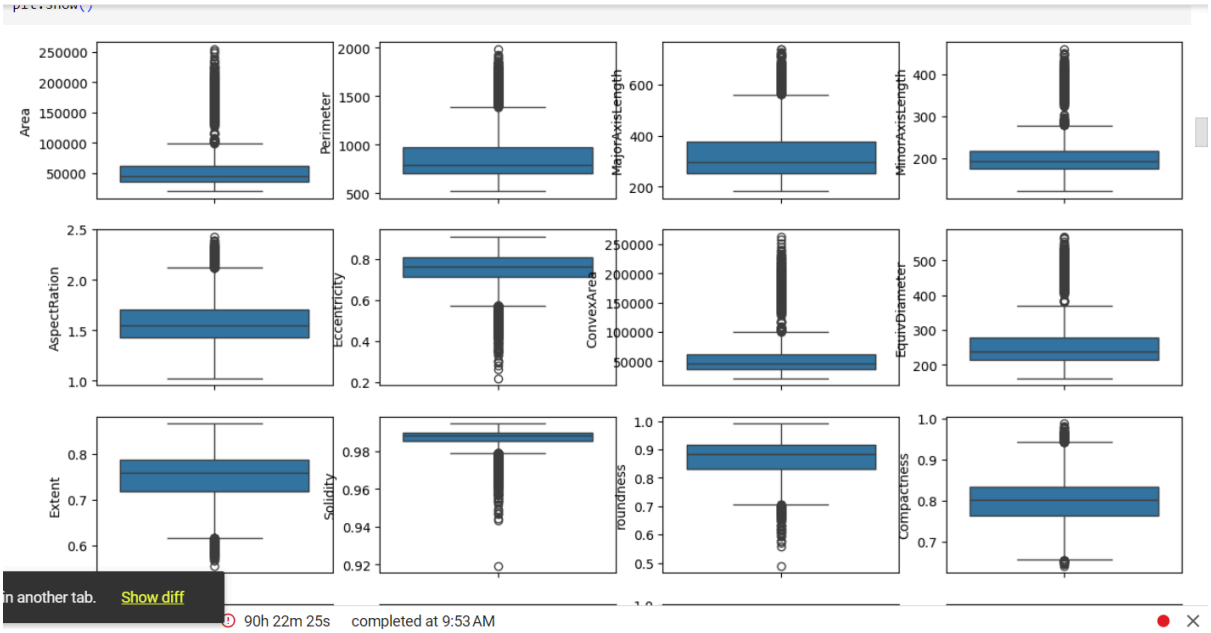
SNO	Feature Name	Data Type
1	Area	int
2	Perimeter	Float
3	MajorAxisLength	Float
4	MinorAxisLength	Float
5	AspectRation	Float
6	Eccentricity	Float
7	ConvexArea	int
8	EquivDiameter	Float
9	Extent	Float
10	Solidity	Float
11	Roundness	Float
12	Compactness	Float
13	ShapeFactor1	Float
14	ShapeFactor2	Float

15	ShapeFactor3	Float
16	ShapeFactor4	Float

The dataset doesn’t contain any Nan values or duplicates but there are outliers in each feature.

Stage 2: Data Preparation

a) Exploratory Data Analysis: The dataset doesn’t contain any Nan values or duplicates but there are outliers in each feature.

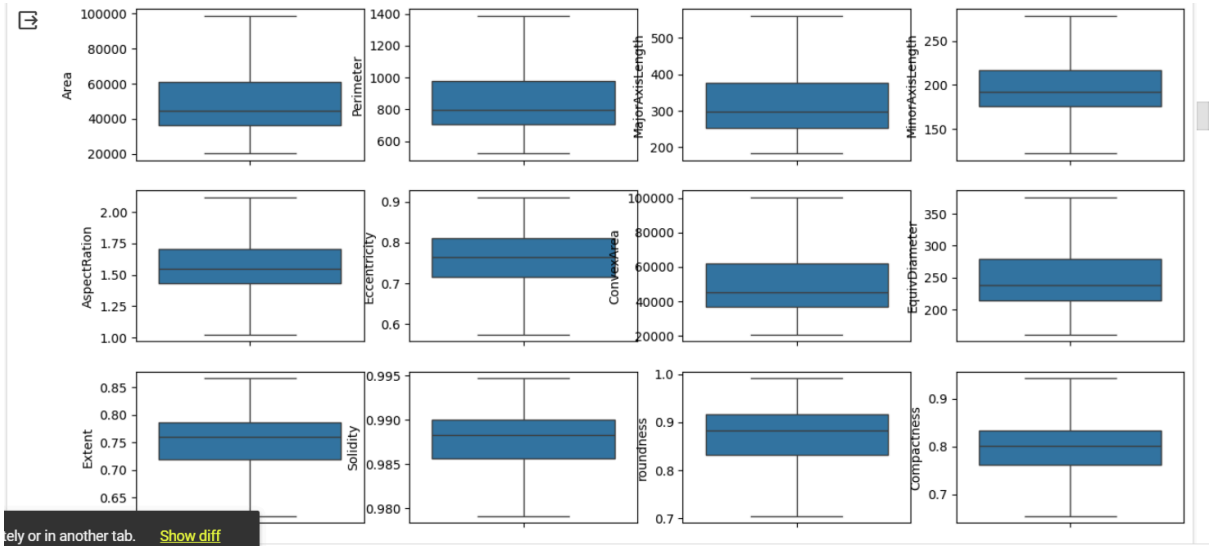


S No	Type	Feature Names	Observation
1	Missing Values	-	
2	Duplicates	-	
3	Outliers	All features	
4	Distributions	-	

b) Data Cleaning/wrangling:

Each feature's outliers are eliminated using the IQR method's clipping procedure. Every outlier exceeding the upper boundary is substituted with the upper

boundary's value, and every outlier falling below the lower boundary is substituted with the lower boundary's value.



c) Feature Selection:

I have determined how many informative features are needed to categorise the dry beans by displaying a table that shows the relationship between each feature and the class variable.

	Area	Perimeter	MajorAxisLength	MinorAxisLength	AspectRation	Eccentricity	ConvexArea	EquivDiameter	ShapeFactor1	ShapeFactor2	Sha
7926	40674	748.346	275.777468	188.399378	1.463792	0.730271	41111.0	227.569210	0.006780	0.001939	
13111	37783	725.097	276.567189	174.202492	1.587619	0.776697	38182.0	219.332646	0.007320	0.001786	
8063	41489	797.126	307.077572	173.107997	1.773907	0.825961	42189.0	229.837846	0.007401	0.001433	
1125	39859	724.609	255.684610	198.752850	1.286445	0.629086	40202.0	225.277729	0.006415	0.002385	
803	38145	762.943	256.113302	190.105434	1.347217	0.670100	38876.0	220.380858	0.006714	0.002271	
...	...	...	...	...	...	...	...	...	...	...	
13531	41221	752.069	280.157423	187.594468	1.493420	0.742719	41590.0	229.094320	0.006796	0.001875	
477	36186	685.681	236.999293	194.915156	1.215910	0.574120	36484.0	214.647260	0.006549	0.002718	
2086	52104	906.358	316.915111	209.604091	1.511970	0.750043	53108.0	257.567221	0.006082	0.001637	
6279	52107	1038.993	393.245196	169.439521	2.119312	0.902412	53645.0	257.574636	0.007547	0.000857	
2869	72837	1052.169	371.910382	250.023336	1.487503	0.740308	73876.0	304.530702	0.005106	0.001416	

10888 rows × 11 columns

S NO	Removed Feature Name	Reason	Test Performed
1	Extent	Almost all the classes are similar.	Table
2	Solidity	Almost all the	Table

		classes are similar.	
3	ShapeFactor4	Almost all the classes are similar.	Table
4	roundness	Almost all the classes are similar.	Table
5	Compactness	Almost all the classes are similar.	Table

Stage 3: Model Building:

I utilised the distance-based K closest neighbour algorithm to categorise dry beans. I separated the dataset into test and training data first. After that, I uniformized the scale of every feature by using MinMaxScaler. Next, using the StratifiedKFold method, I divided the train data into five folds by dividing it into train and cross-validation datasets. I trained the model using the K closest Neighbours technique for classification.

S NO	**Type of Problem**	**Approach**	**Algorithm Name**
1	Classification	Distance Based	KNN

Stage 4: Model Training:

Using K=1 to 40 values, I attempted to determine the optimal range of K values without overfitting or underfitting by graphing the train and test data against K. By predicting the class label for cross validation data, I was able to determine the model's performance within the range of K values that I had collected.

S No	**Algorithm Name**	**Hyper-parameter tuning**	**Metric used for Evaluation**
1	KNN	1-10	Accuracy

Stage 5: Model Evaluation:

To evaluate the performance of the model, I used accuracy as my performance metric because there is a high variation in accuracy value for each K value. The rest of performance metrics are producing almost similar values but there is a small difference for each value of K. Hence I used accuracy as my performance metric.

S NO	**Algorithm Name**	**Hyper parameter**	**accuracy**	
1	KNN	211	0.8905618802791039	
2	KNN	7	0.9085567388909291	

Conclusion:

Upon building numerous KNN models and adjusting the hyperparameter, the model exhibits good performance at K=7. When compared to other K values, the accuracy is good and the accuracy is incredibly low at K=7, which is the model's performance indicator