CSE343/ECE343: Machine Learning **Name:** *Srishti Singh*
Assignment-1 Linear Regression, Logistic Regression, Naive Bayes
September 2022 **rollno:** *2020409*

1. Section A (Theoretical)

(a) For simple linear regression,…variables respectively. Prove it.

The Least Squares Regression Line is the line that makes the vertical distance from the data points to the regression line as small as possible.
Suppose a training set of n observation is given - $(x_i, y_i)(i = 1, 2 \dots n)$
We can define the best fit line as $\hat{Y_i} = a + b.X_i$
In this equation $\hat{Y_i}$ is termed as dependent study variable and X_i is termed as independent study variable. Minimum cost line seeks to minimize the square error cost function to provide the best fit for the points.

$$J(a,b) = \sum_{i=1}^{n}(Y_i - \hat{Y_i})^2$$

$$J(a,b) = \sum_{i=1}^{n}(Y_i - a - b.X_i)^2$$

To minimize cost function we will take the derivative of $J(a,b)$ wrt a, b and equate it to 0.

$$\frac{\partial J(a,b)}{\partial a}[\sum_{i=1}^{n}(Y_i - a - b.X_i)^2] = 0$$

$$-2\sum_{i=1}^{n}(Y_i - a - b.X_i) = 0$$

$$\sum_{i=1}^{n}Y_i - \sum_{i=1}^{n}a - \sum_{i=1}^{n}b.X_i = 0$$

$$\sum_{i=1}^{n}Y_i - \sum_{i=1}^{n}b.X_i = n.a$$

$$.\frac{\sum_{i=1}^{n}Y_i}{n} - \frac{b\sum_{i=1}^{n}X_i}{n} = a$$

$$\text{Let } \frac{\sum_{i=1}^{n}Y_i}{n} = \bar{Y} \text{ and } \frac{\sum_{i=1}^{n}X_i}{n} = \bar{X}$$

$$\text{So } \bar{Y} = a + b\bar{X}$$

This shows that the arithmetic mean of dependent and independent variables will always pass through the Least Squares Regression Line.

(b) Let us suppose if two variables have a high ... answer with the help of an example.

Example -
Suppose a person is trying to sell a house. The price of the house is strongly correlated to the color of the house because beautiful color makes the house more attractive. Similarly the avalaibility of a garden/balcony also positively affects the price and they are strongly related. However the the availability of the garden and the color of the house are not dependent on each other hence they are not strongly correlated. Correlation of between the two variables is independent of the third variable which they are strongly correlated to however in some situation it may also be dependent so therefore the answer to this questions depends on the variables rather than maths.

(c) (2 marks) Provide proof of the weak law of large numbers (LLN). Provide a pseudo-code to illustrate the weak LLN assuming some distribution for the random variable.

let $X_1, X_2 \dots X_n$ be i.i.d random variables with some finite Expected Value $EX_i = \mu < \infty$
The Weak law of large number states that

$$\lim_{n \to \infty} (P|\bar{X} - \mu| >= \epsilon) = 0 \tag{1}$$

Assumption Var(x) is finite

$$Var(X) = \sigma^2 = finite \tag{2}$$

Chebyshev's inequality states that - Probability that the difference between Random variable X and $\bar{X}$(mean) is some small value k is less than or equal to the variance of X divided be square of said small value k.

$$Pr(|x - \mu| >= k\sigma) <= \frac{1}{k^2} \tag{3}$$

x = random variable, σ = standard deviation
μ = expected value, k = number of standard deviations

By using Chebyshev's inequality we can write -

$$P(|X - \mu| >= k\sigma) <= \frac{Var(\bar{X})}{\epsilon^2} \tag{4}$$

Var can be simplified to -

$$\begin{aligned}
Var(\bar{X}) &= \frac{Var(X_1 + X_2 \dots X_n)}{n^2} && \text{since } Var(aX) = a^2 Var(X) \\
&= \frac{Var(X_1) + Var(X_2) \dots Var(X_n)}{n^2} && \text{since the } X_i's \text{ are independent} \\
&= \frac{nVar(X)}{n^2} && \text{since } Var(X_i) = Var(X) \\
&= \frac{Var(X)}{n}
\end{aligned}$$

Now putting this into equation (4) we will get -

$$\frac{Var(\bar{X})}{\epsilon^2} = \frac{Var(X)}{n\epsilon^2}$$

Hence we will finally get the equation -

$$P(|X - \mu| >= k\sigma) <= \frac{Var(X)}{n\epsilon^2} \tag{5}$$

when n approaches infinity $\frac{Var(X)}{n\epsilon^2}$ will be zero hence

$$P(|X - \mu| >= k\sigma) <= 0 \tag{6}$$

Hence Proved.
This is the pseudocode to flip a coin. The graphs are showing that as the number of samples increase the average value of dice (0.5*0 + 0.5*1) is stabilizing to 0.5.

```
def Flip_Coin(n):
    #List to store the result of coin flip
  result = []
  for i in range(1,n+1):
      result.append(random.choice([0,1]))
  return result

def results(n):
  result=   Flip_Coin(n)
  averages = []
  _cumsum = np.cumsum(result)
  for index in range(len(_cumsum)):
    averages.append(_cumsum[index]/(index+1))
  return averages
```

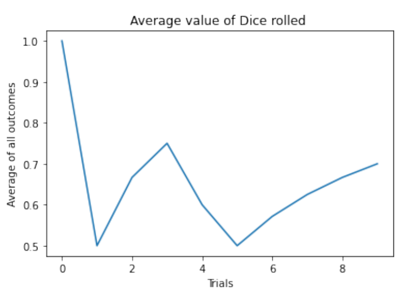


Figure 1: n=10

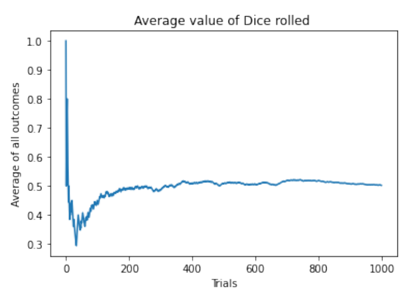


Figure 2: n=1000

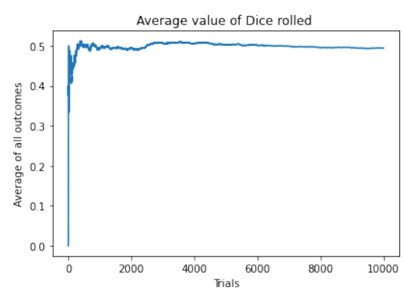


Figure 3: n=10000

(d) (3 marks) Derive the Maximum A Posterior (MAP) solution for linear regression.(assuming Gaussian prior distribution of the weights).

Assuming Gaussian Prior Distribution -

$$P(w) = \mathcal{N}(w|0,\, \lambda^{-1}I) = \frac{1}{2\pi^{D/2}} exp\left(\frac{-\lambda}{2} w^T w\right) \tag{7}$$

MAP will be -

$$P(w|D) = \frac{P(D|w)P(w)}{P(D)}$$

Log Posterior Probability -

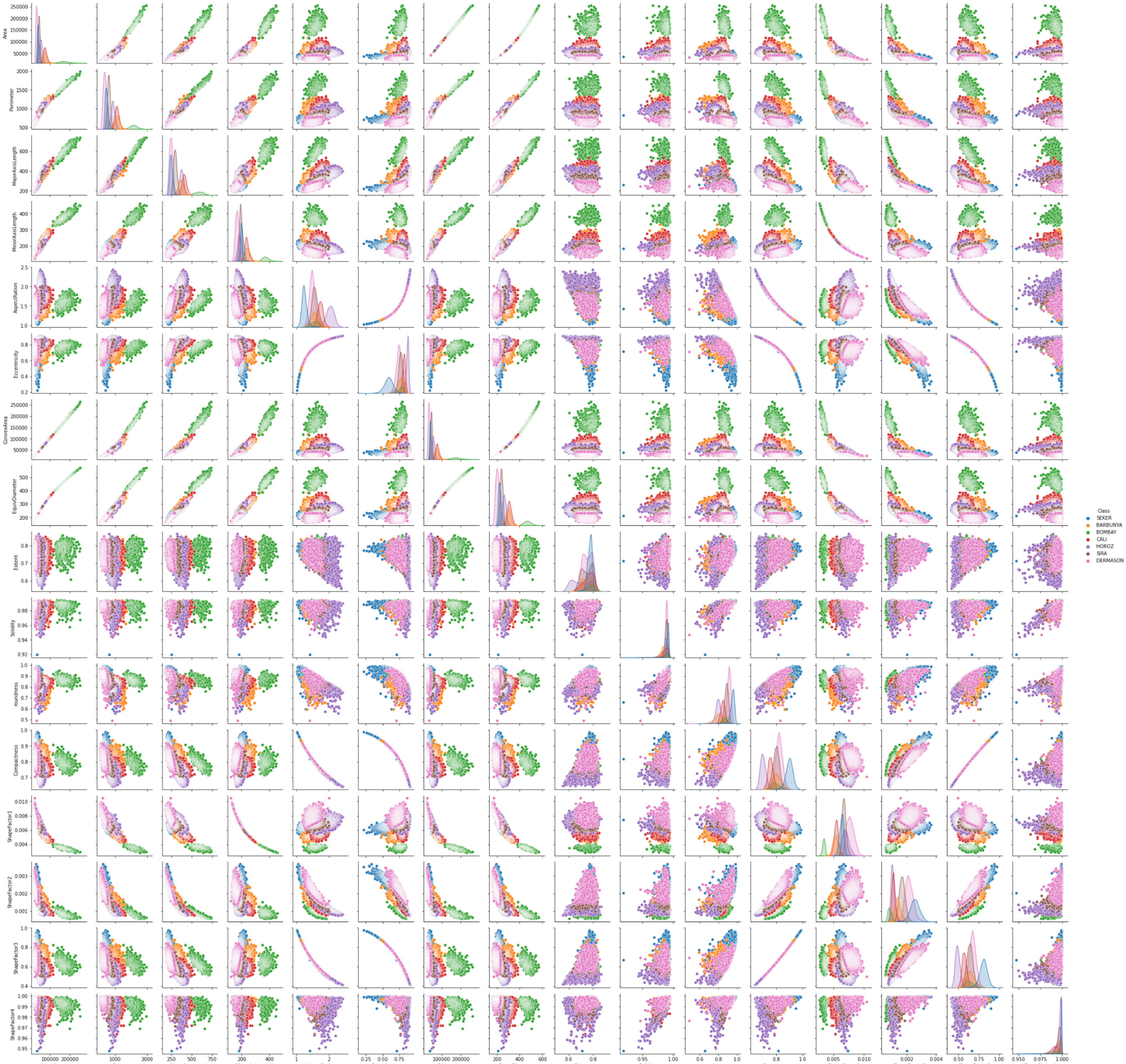
$$\log P(w|D) = \log \frac{P(D|w)P(w)}{P(D)} = \log P(D|w) + \log P(w) - \log P(D)$$

Maximum a Posterior Solution-

$$\begin{aligned}
\hat{w}_{MAP} &= \arg\max_w \log P(w|D) \\
&= \arg\max_w \log P(D|w) + \log P(w) - \log P(D) \\
&= \arg\max_w \log P(D|w) + \log P(w) \\
&= \arg\max_w \log P(D|w) + \log \frac{1}{2\pi^{D/2}} exp\left(\frac{-\lambda}{2} w^T w\right) && \text{using (7)} \\
&= \arg\max_w \log P(D|w) + \frac{-D}{2} \log 2\pi - \frac{\lambda}{2} w^T w && \text{substituting } \log P(D|w) \text{from MLE} \\
&= \arg\max_w \sum_{i=1}^{n} \{ \frac{-\log(2\pi\sigma^2)}{2} - \frac{(y_i - w^T x_i)^2}{2\sigma^2} \} - \frac{\lambda}{2} w^T w && \text{removing constant term} \\
\hat{w}_{MAP} &= \operatorname*{argmin}_w \frac{1}{2\sigma^2} \sum_{i=1}^{n} (y_i - w^T x_i)^2 + \frac{\lambda}{2} w^T w && \text{changing max to min}
\end{aligned}$$

This is know as regularized version of MSE.
Hence Proved.

Section C


Area
Perimeter
MajorAxisLength
MinorAxisLength
AspectRation
Eccentricity
ConvexArea
EquivDiameter
Extent
Solidity
roundness
Compactness
ShapeFactor1
ShapeFactor2
ShapeFactor3
ShapeFactor4
Class
SEKER
BARBUNYA
BOMBAY
CALI
HOROZ
SIRA
DERMASON

(a) Analysis -

When y increases as x increases we say that two features are strongly correlated. Some strongly related features are -

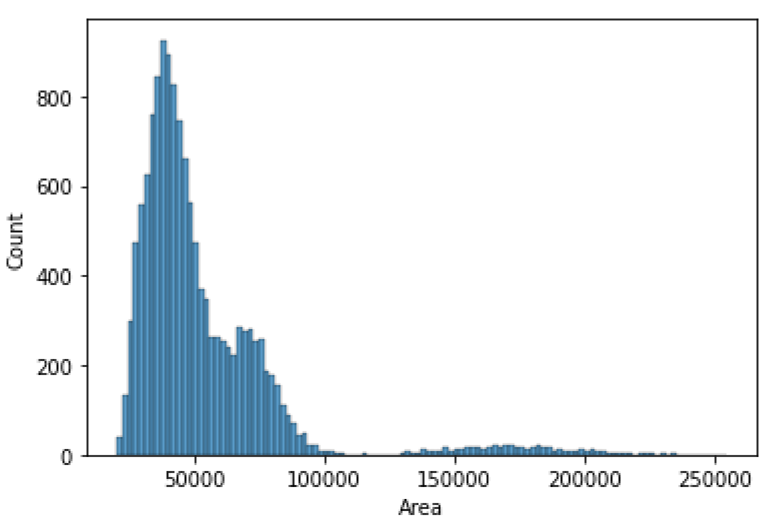
1. Area & Perimeter
2. Area & Convex area
3. Area & Equidiameter
4. Compactness & Shapefactor3
5. Equidiamete & Convex area etc.

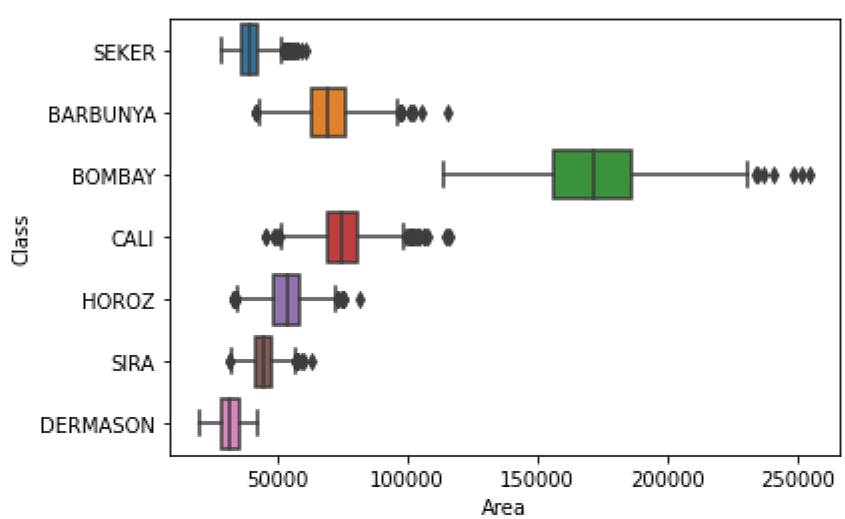
When y decreases as x increases we say that two features are negatively correlated. Some negatively related features are -

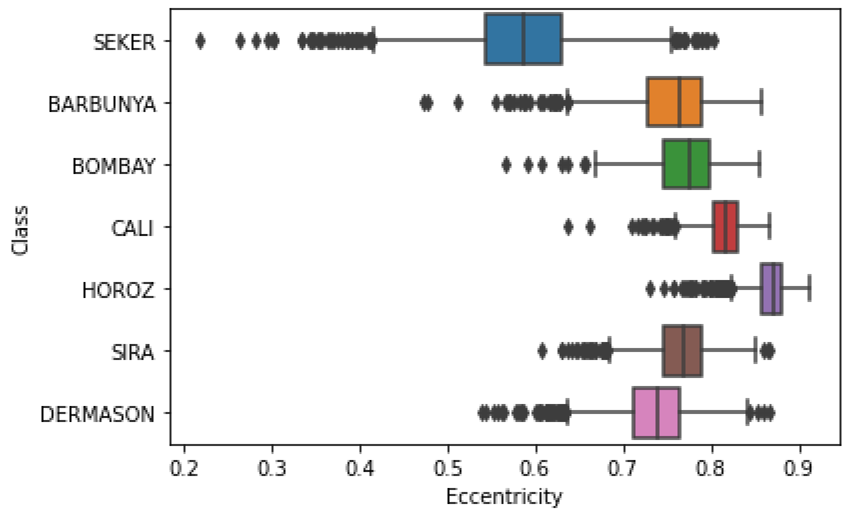
1. Shapefactor1 & Area
2. Shapefactor1 & Perimeter
3. Shapefactor1 & MajorAxis
4. Shapefactor1 & Convex Area
5. Shapefactor3 & Aspectratio etc

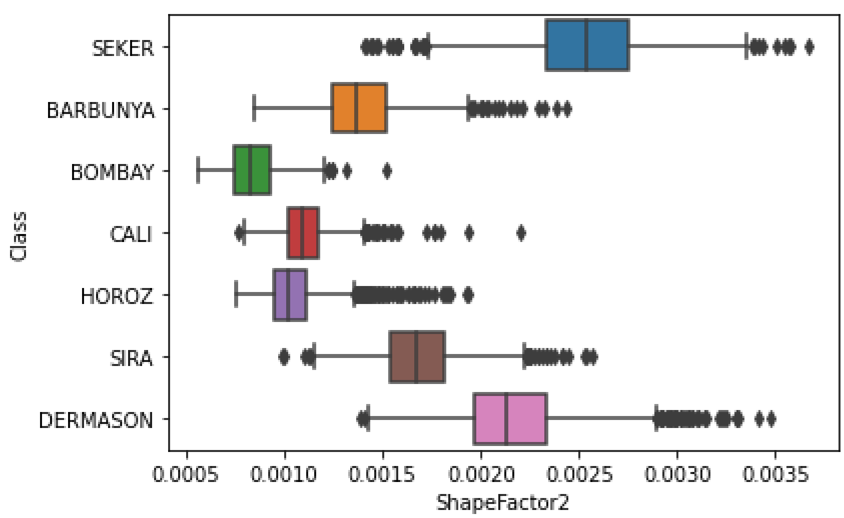
(b)Insights-

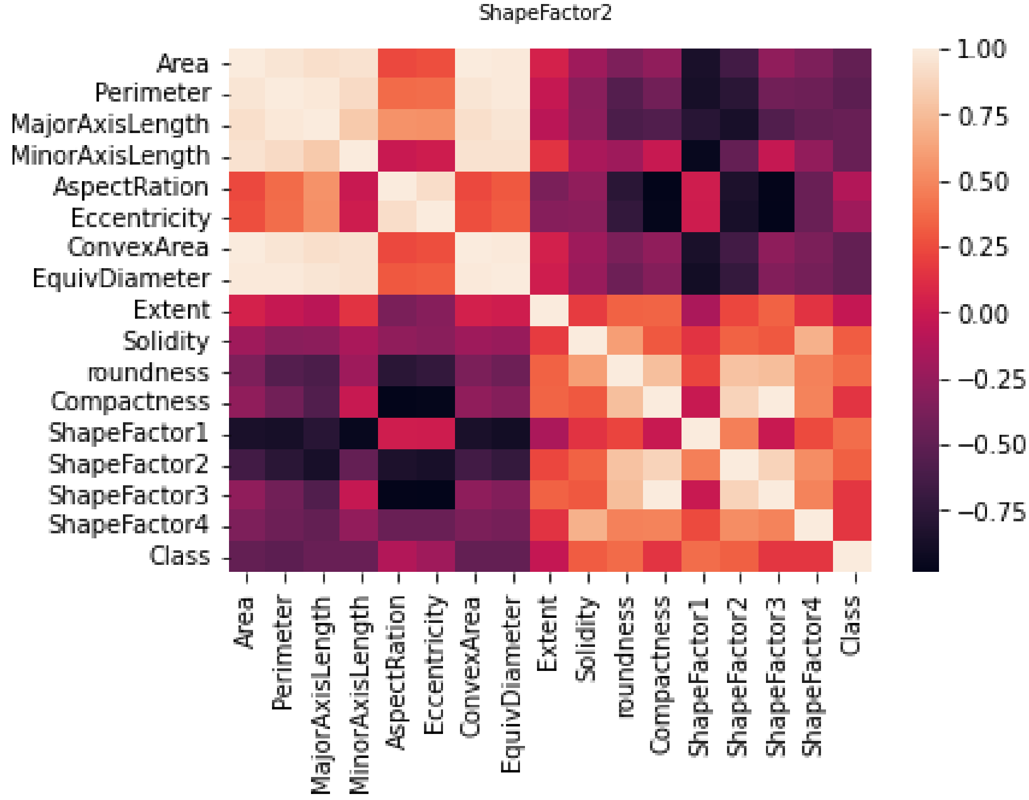
- Area of most of the set lie in range 0 to 100000 (histplot)
- Bombay has Maximum area and Dermason has minimum area (box plot)
- Horoz has maximum eccentricity
- eccentricity and AspectRation are strongly correlated with shapefactor3 (heatmap)
- eccentricity and AspectRation are strongly correlated with compactness (heatmap)
- equidiameter and convex area are weakly correlated (heatmap)
- shapefactor2 has a lot of boundary/stray points(boxplot)






SEKER
BARBUNYA
BOMBAY
CALI
HOROZ
SIRA
DERMASON
Class
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
Eccentricity


SEKER
BARBUNYA
BOMBAY
CALI
HOROZ
SIRA
DERMASON
Class
0.0005
0.0010
0.0015
0.0020
0.0025
0.0030
0.0035
ShapeFactor2


Area
Perimeter
MajorAxisLength
MinorAxisLength
AspectRation
Eccentricity
ConvexArea
EquivDiameter
Extent
Solidity
roundness
Compactness
ShapeFactor1
ShapeFactor2
ShapeFactor3
ShapeFactor4
Class
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75

```
[ ] df.isnull().sum()
```

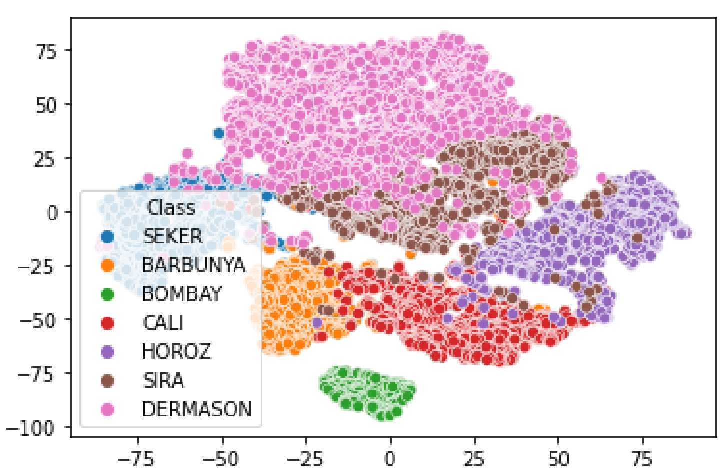
```
Area               0
Perimeter          0
MajorAxisLength    0
MinorAxisLength    0
AspectRation       0
Eccentricity       0
ConvexArea         0
EquivDiameter      0
Extent             0
Solidity           0
roundness          0
Compactness        0
ShapeFactor1       0
ShapeFactor2       0
ShapeFactor3       0
ShapeFactor4       0
Class              0
dtype: int64
```

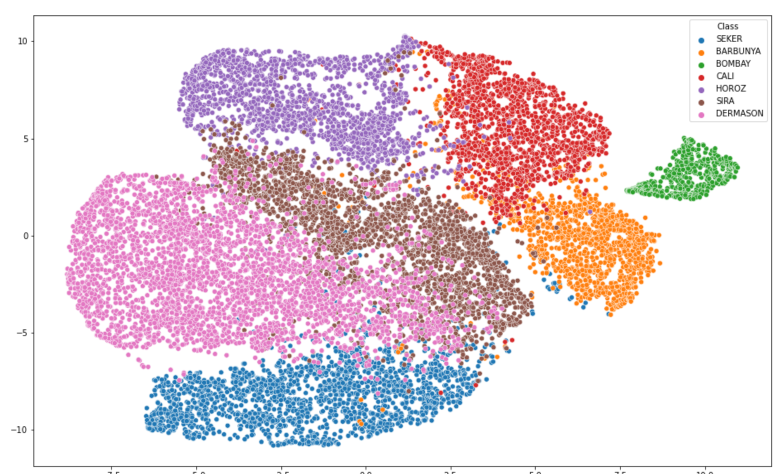
```
[ ] df.isnull().values.any()
```

```
False
```

(c) Tsne

Data is pretty seperable at for distiguishing green red and orange however there are some merging points between blue pink brown and purple.





(d) Gaussian Naive Bayes shows higher accuracy and precision. On Standardizing the data the accuracy becomes almost 93 percent which means gaussian is better for this model.

```
Gaussian
Accuracy in % : 77.12082262210797
precision in % :  77.28148416428769
recall score in % :  [ 48.59437751 100.          78.85304659  83.70165746  78.27225131
  73.58490566  77.87769784]
F1 score in % :  77.0235418829796
              precision    recall  f1-score   support

           0       0.65      0.49      0.56       249
           1       0.99      1.00      1.00       109
           2       0.67      0.79      0.72       279
           3       0.87      0.84      0.85       724
           4       0.77      0.78      0.78       382
           5       0.69      0.74      0.71       424
           6       0.77      0.78      0.78       556

    accuracy                           0.77      2723
   macro avg       0.77      0.77      0.77      2723
weighted avg       0.77      0.77      0.77      2723

Number of mislabeled points out of a total 2723 points : 623
```

```
Bernoulli
Accuracy in % : 26.588321704002936
precision in % :  3.79833167200042
recall score in % :  [  0.   0.   0. 100.   0.   0.   0.]
F1 score in % :  6.001077541547516
              precision    recall  f1-score   support

           0       0.00      0.00      0.00       249
           1       0.00      0.00      0.00       109
           2       0.00      0.00      0.00       279
           3       0.27      1.00      0.42       724
           4       0.00      0.00      0.00       382
           5       0.00      0.00      0.00       424
           6       0.00      0.00      0.00       556

    accuracy                           0.27      2723
   macro avg       0.04      0.14      0.06      2723
weighted avg       0.07      0.27      0.11      2723
```

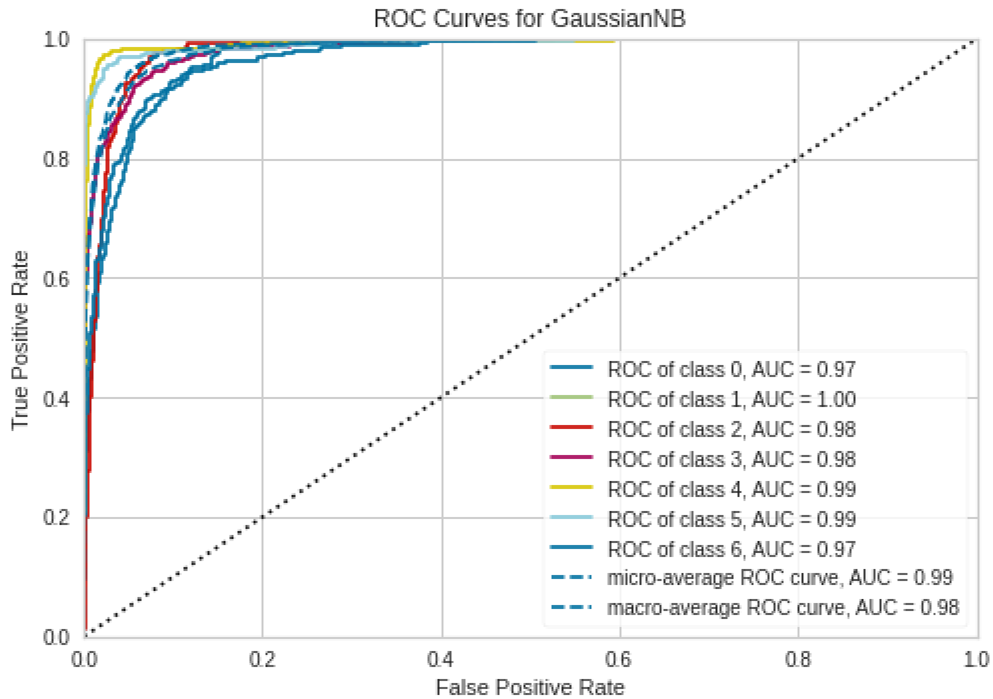
(e) I got the best results in n = 6

```
Accuracy in % : 90.08446566287184
precision in % :  91.80653717072276
recall score in % :  [ 89.27203065 100.          92.42902208  84.50074516  94.60784314
  93.22033898  88.05970149]
F1 score in % :  91.69060681216553
              precision    recall  f1-score   support

           0       0.92      0.89      0.91       261
           1       1.00      1.00      1.00       117
           2       0.91      0.92      0.92       317
           3       0.93      0.85      0.88       671
           4       0.91      0.95      0.93       408
           5       0.96      0.93      0.95       413
           6       0.79      0.88      0.83       536

    accuracy                           0.90      2723
   macro avg       0.92      0.92      0.92      2723
weighted avg       0.90      0.90      0.90      2723
```

(f) Since the area under the curves is high this shows the training sets and samples that we chose were great and fit the model nicely.



(f) Logistic has a better accuracy and is more suitable for this type of dataset as this is sort of a classification problem.

Logistic Regression-

```
Accurarcy -  91.97766676461944
precision in % :  93.56905342752555
recall score in % :  [ 89.91825613 100.          96.25935162  91.87643021  94.06779661
  94.73684211  85.62874251]
F1 score in % :  93.3652004648678
              precision    recall  f1-score   support

           0       0.96      0.90      0.93       367
           1       1.00      1.00      1.00       127
           2       0.92      0.96      0.94       401
           3       0.91      0.92      0.91       874
           4       0.96      0.94      0.95       472
           5       0.95      0.95      0.95       494
           6       0.85      0.86      0.85       668

    accuracy                           0.92      3403
   macro avg       0.94      0.93      0.93      3403
weighted avg       0.92      0.92      0.92      3403
```

```
Gaussian
Accuracy in % : 77.12082262210797
precision in % :  77.28148416428769
recall score in % :  [ 48.59437751 100.          78.85304659  83.70165746  78.27225131
  73.58490566  77.87769784]
F1 score in % :  77.0235418829796
              precision    recall  f1-score   support

           0       0.65      0.49      0.56       249
           1       0.99      1.00      1.00       109
           2       0.67      0.79      0.72       279
           3       0.87      0.84      0.85       724
           4       0.77      0.78      0.78       382
           5       0.69      0.74      0.71       424
           6       0.77      0.78      0.78       556

    accuracy                           0.77      2723
   macro avg       0.77      0.77      0.77      2723
weighted avg       0.77      0.77      0.77      2723

Number of mislabeled points out of a total 2723 points : 623
```