Exploratory Data Analysis

Description of data at a glance

We will use this dataframe for further analysis

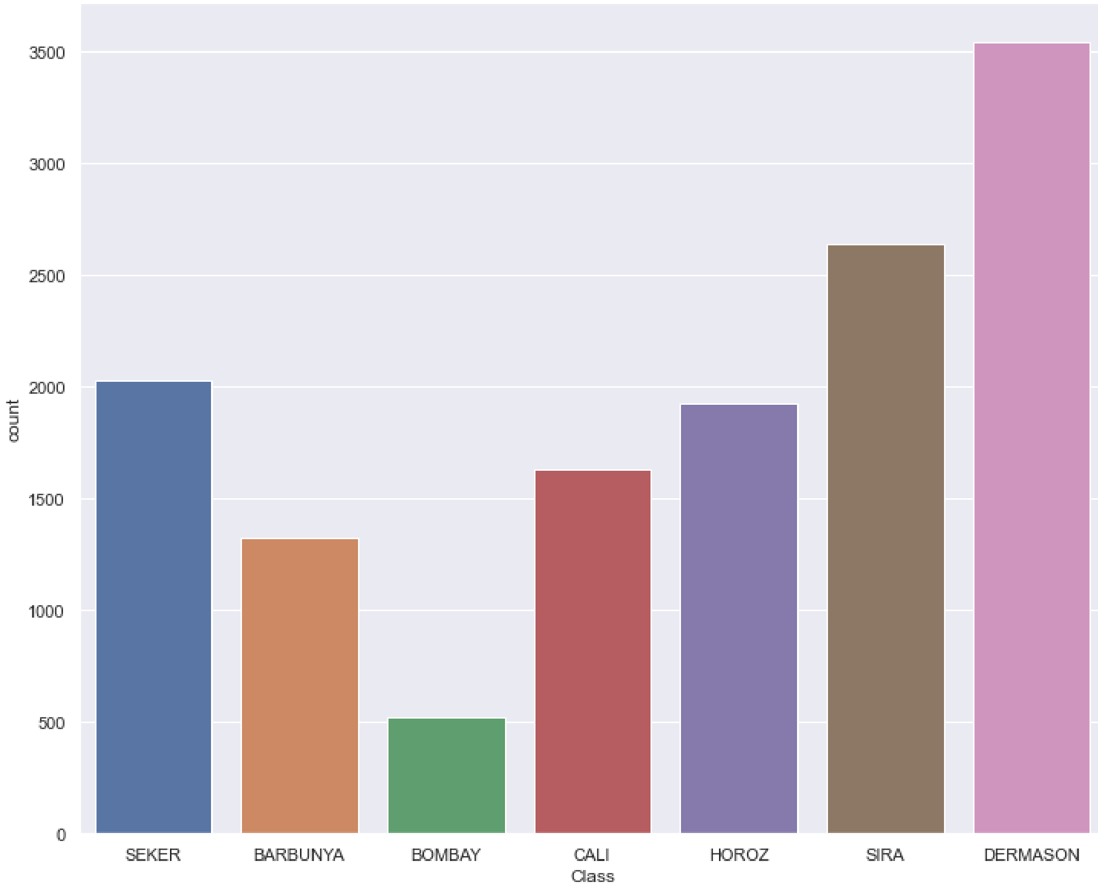
A brief overview of the dataframe

```
RangeIndex: 13611 entries, 0 to 13610
Data columns (total 17 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   Area             13611 non-null  int64
 1   Perimeter        13611 non-null  float64
 2   MajorAxisLength  13611 non-null  float64
 3   MinorAxisLength  13611 non-null  float64
 4   AspectRation     13611 non-null  float64
 5   Eccentricity     13611 non-null  float64
 6   ConvexArea       13611 non-null  int64
 7   EquivDiameter    13611 non-null  float64
 8   Extent           13611 non-null  float64
 9   Solidity         13611 non-null  float64
 10  roundness        13611 non-null  float64
 11  Compactness      13611 non-null  float64
 12  ShapeFactor1     13611 non-null  float64
 13  ShapeFactor2     13611 non-null  float64
 14  ShapeFactor3     13611 non-null  float64
 15  ShapeFactor4     13611 non-null  float64
 16  Class            13611 non-null  object
dtypes: float64(14), int64(2), object(1)
memory usage: 1.8+ MB
```

- We have 16 features, 12 dimensional and 4 shape features
- The `Class` column contains the Classes
- We have 13,611 rows each corresponding to 16 features per bean
- We have got 5 different classes: 'SEKER', 'BARBUNYA', 'BOMBAY', 'CALI', 'HOROZ', 'SIRA', 'DERMASON'

Analysisng the Classes

```
_ = sns.countplot(data=df, x='Class')
```

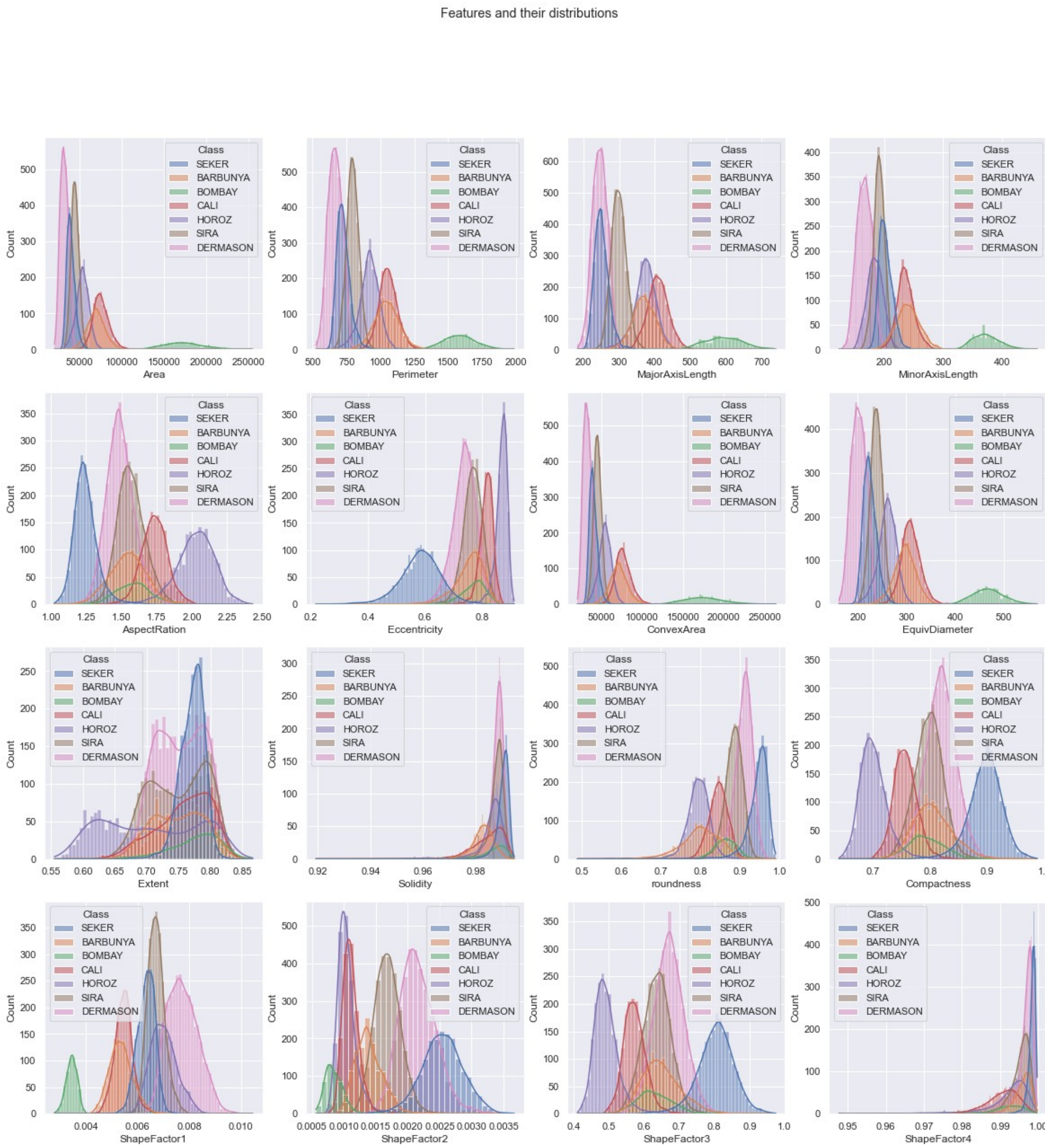


Obvservation

- We have got 5 classes and above are the counts of the classes. As, we can see that the majority class is `DERMASON` and minority one is `BOMBAY`. The data is imbalanced as `BOMBAY` has only 500 examples where as `DERMASON` has 3500 examples.

Analysing the features

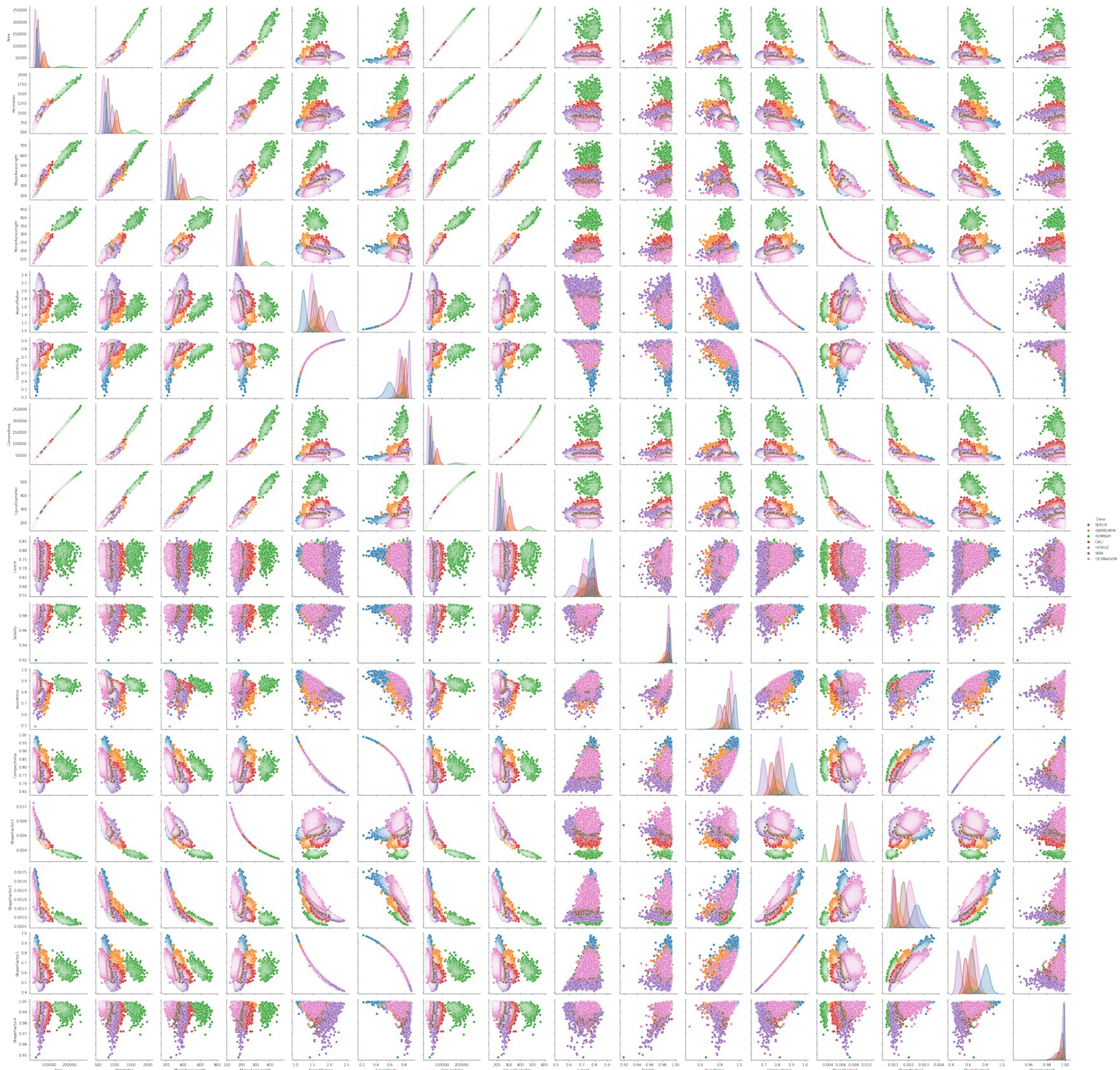
Univariate Analysis



Obvservation

- BOMBAY class can be differentiated easily using any feature
- The other classes have a lot of overlap and are not easy to distinguish

Bivariate Analysis

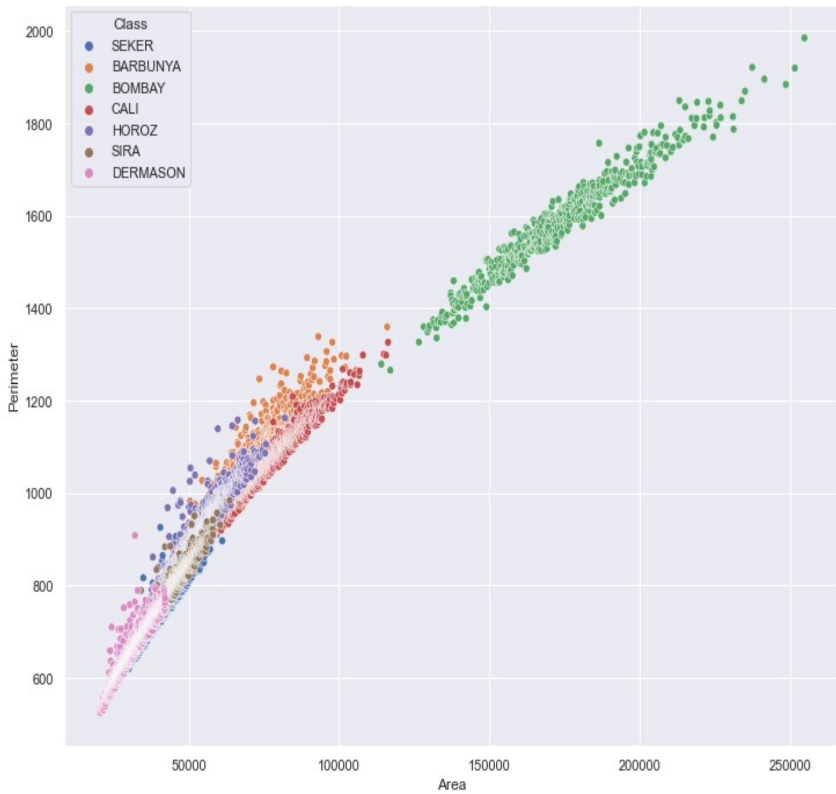


Since there are a lot of features, let's look at the pair plot first and then we can progress

obvservation

- So, it's kinda cumbersome, but still it gives us some details about it our data
- The green colored points are points belonging to `BOMBAY` our minority class. It seems any feature is good enough for separating BOMBAY from other classe.
- We can't really say the same for other classes

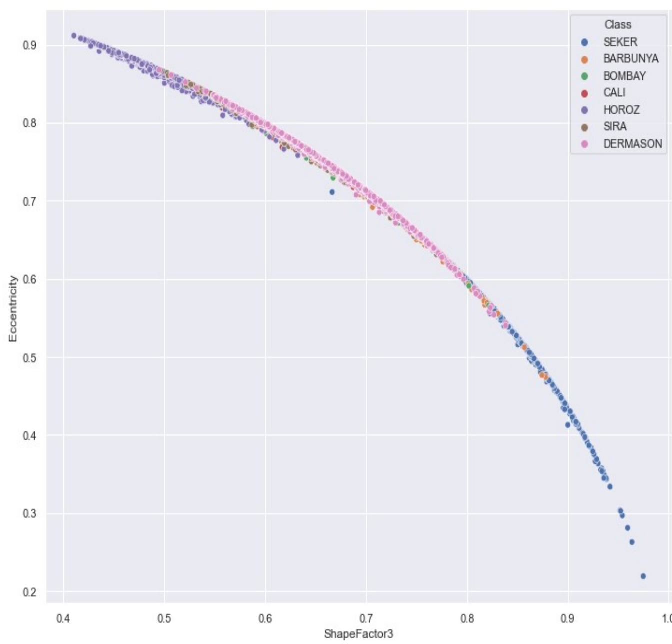
We can zoom in on some of the plots and see up close for ourselves that BOMBAY is easy to separate



Class
SEKER
BARBUNYA
BOMBAY
CALI
HOROZ
SIRA
DERMASON
Perimeter
Area
2000
1800
1600
1400
1200
1000
800
600
50000
100000
150000
200000
250000

Class
SEKER
BARBUNYA
BOMBAY
CALI
HOROZ
SIRA
DERMASON
ShapeFactor2
ShapeFactor1
0.0035
0.0030
0.0025
0.0020
0.0015
0.0010
0.0005
0.003
0.004
0.005
0.006
0.007
0.008
0.009
0.010

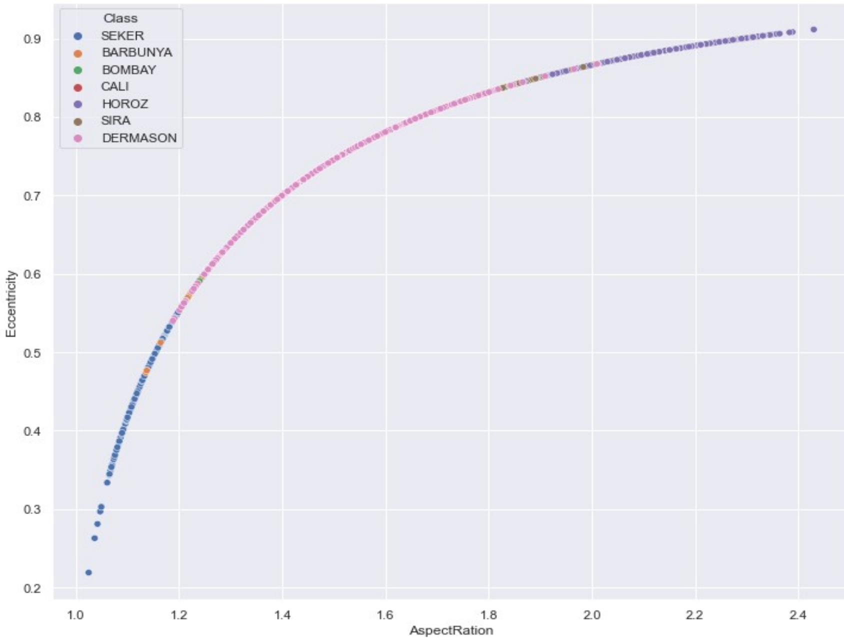
Now, lets zoom in on some features which are highly correlated to each other

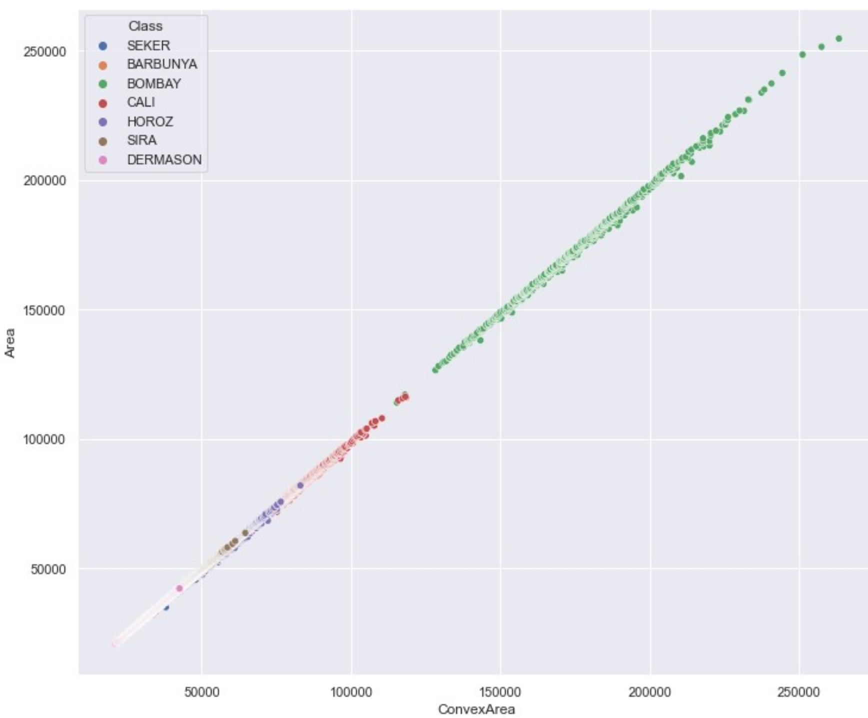


Obvservation

- ShapeFactor3 and Eccentricity are high negative correlation, they seem to be perfectly lining up

Few other features with high negative and positive correlation



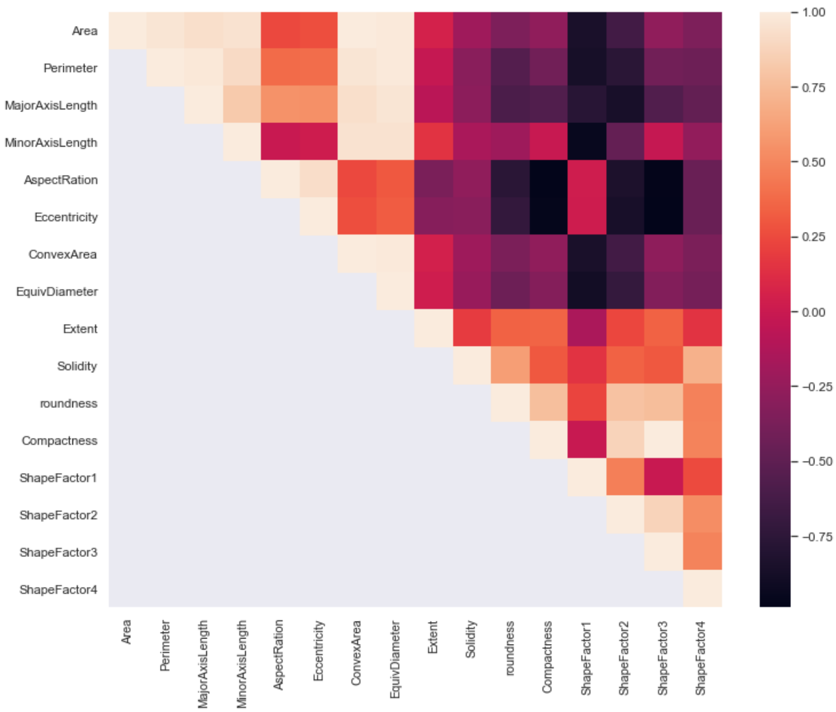


Obvservation

- This is pretty interesting to look at, Area and ConvexArea seem to be the exact same features. Makes sense as ConvexArea approximates Area to the closest convex polygon

Let's Move on to correlation analysis

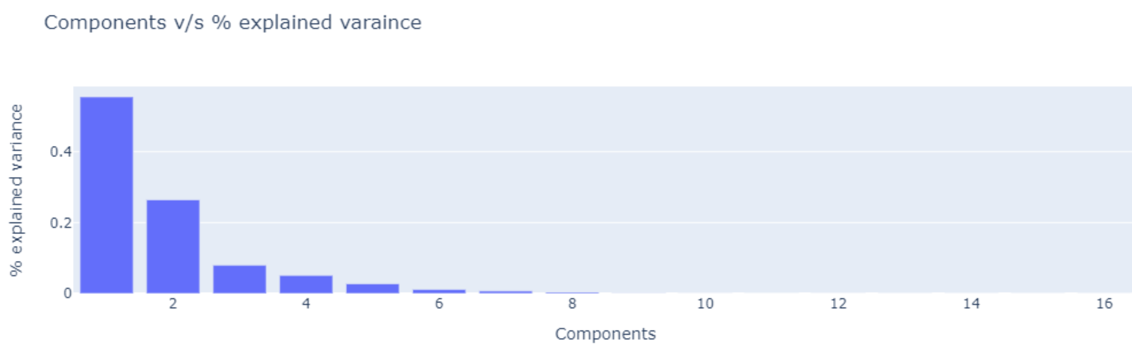
Correlation Analysis



Obvservation

- As, we can see that most of our features are highly correlated either negatively or positively.
- My hypothesis is even if we use very less features, we will still be able to descibe our data well.

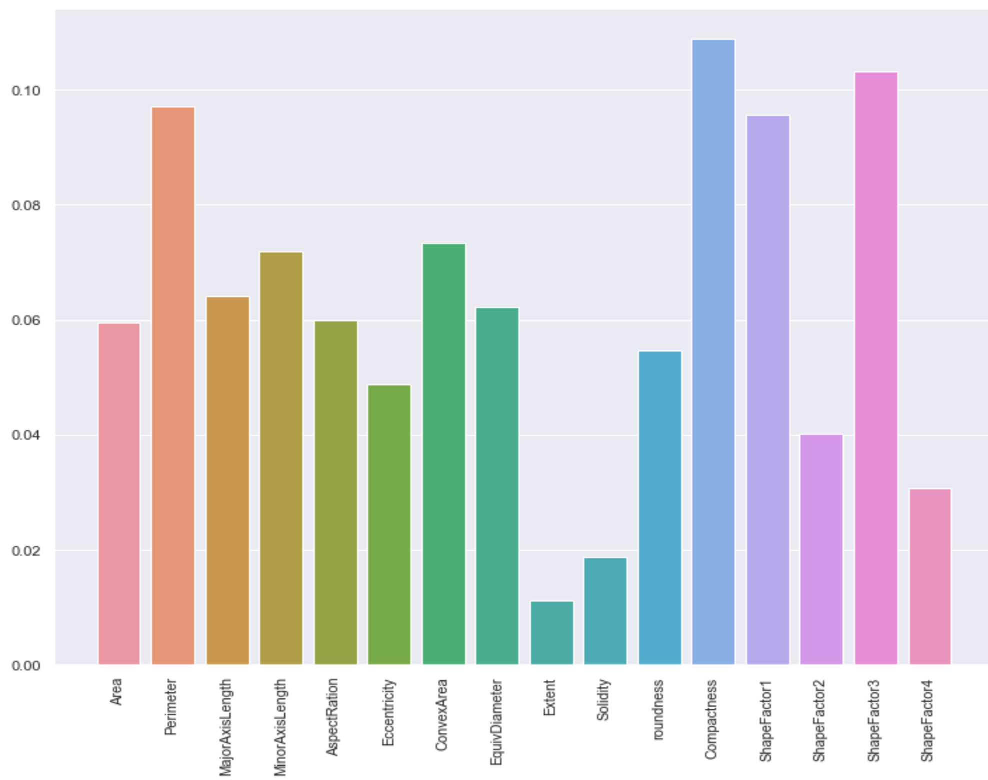
Let's use PCA and see if that holds true. The idea of PCA is simple — reduce the number of variables of a data set, while preserving as much information as possible.



Obvservation

- It's interesting to see that just the first 5 components are good enough to explain 96 % of the data. If we take just 8 out of the 16 components, we can explain the whole data
- But PCA is like not so good for interpretebility. Nonetheless, it kind of validates my hypothesis

Feature importances



Obvservation

- `ShapeFactor3`, `Compactness`, `Perimeter` have the highest importances.
- We will use one of them to remove outliers.

Cleaning the data

Dealing with missing values

column	NA count	Null count
Area	0	0
Perimeter	0	0
MajorAxisLength	0	0
MinorAxisLength	0	0
AspectRation	0	0
Eccentricity	0	0
ConvexArea	0	0
EquivDiameter	0	0
Extent	0	0
Solidity	0	0
roundness	0	0
Compactness	0	0
ShapeFactor1	0	0
ShapeFactor2	0	0
ShapeFactor3	0	0

```
|   ShapeFactor4  |    0     |     0      |
|      Class      |    0     |     0      |
+-----------------+----------+------------+
```

obvservation

- As you can see, the dataset is fairly complete with no missing or na values. So, we don't need to deal with them

Checking for negative values

Since all the featues are either dimensional or derived from the dimensional features, thhe values can't be negative. Let's cehck for negative features.

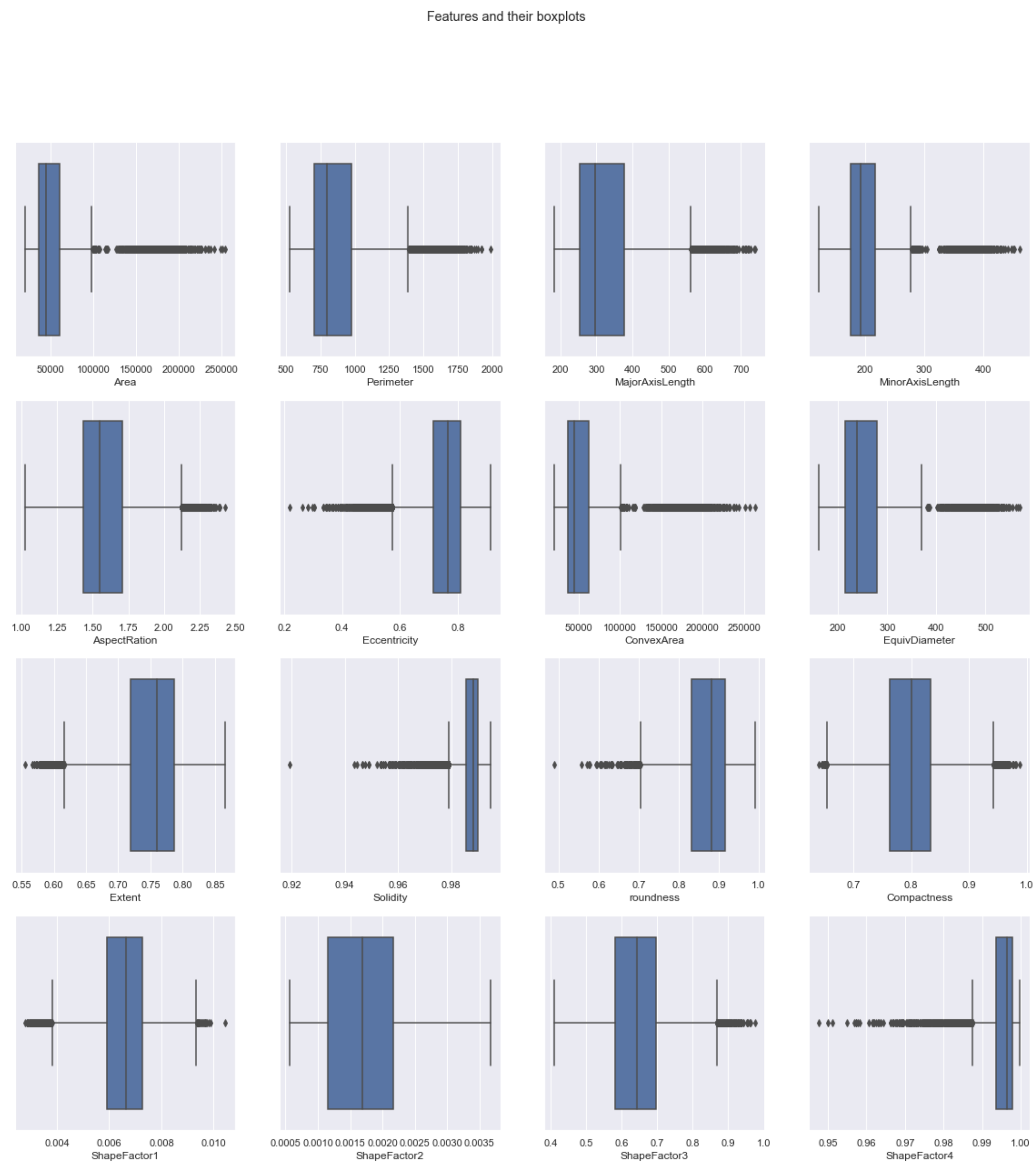
```
Area               0
Perimeter          0
MajorAxisLength    0
MinorAxisLength    0
AspectRation       0
Eccentricity       0
ConvexArea         0
EquivDiameter      0
Extent             0
Solidity           0
roundness          0
Compactness        0
ShapeFactor1       0
ShapeFactor2       0
ShapeFactor3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

Obvservation

- All the columns have positive values which is good as we can now use all rows

Outlier Removal

Let's see the distirbution of the features



Outlier removal using zscore method

column	method	% data retained
roundness	zscore	100.0
Solidity	zscore	100.0
ShapeFactor4	zscore	100.0
Extent	zscore	100.0
Eccentricity	zscore	100.0
ShapeFactor1	zscore	99.993
Compactness	zscore	99.993
ShapeFactor2	zscore	99.963
ShapeFactor3	zscore	99.941
AspectRation	zscore	99.89
MajorAxisLength	zscore	97.678
Perimeter	zscore	97.032
EquivDiameter	zscore	96.584
ConvexArea	zscore	96.451
Area	zscore	96.451
MinorAxisLength	zscore	96.268

Removal of outliers using iqr method

column	method	% data retained
ShapeFactor2	iqr	100.0
roundness	iqr	99.331
Compactness	iqr	99.199
ShapeFactor3	iqr	98.567
Extent	iqr	97.98
MajorAxisLength	iqr	97.215
AspectRation	iqr	96.525
Perimeter	iqr	96.327
EquivDiameter	iqr	96.135
ShapeFactor1	iqr	96.084
ConvexArea	iqr	95.959
Area	iqr	95.952
MinorAxisLength	iqr	95.82
ShapeFactor4	iqr	94.365
Solidity	iqr	94.284
Eccentricity	iqr	93.806

Conclusion

We get a lot of insights from the data:

- The dataset is clean
- Features are highly correlated
- Removal of few features will not impact the performance or interpretability