To Analyse unsupervised learning algorithms on Breast Cancer data set and to perform External and Internal Cluster validation techniques

Problem to be solved

To Apply Gaussian Mixture Model and KMeans Algorithm on the dataset and to find the optimum number of clusters by using different Internal cluster validation techniques and to perform various external cluster validation techniques for 2 cluster models because we have labelled classes for number of clusters = 2

Algorithm to be followed

We have used 2 unsupervised learning algorithms

- **KMeans Clustering** - It is a clustering algorithm that categorizes the items into k groups of similarity. To calculate that similarity, we have used the Euclidean distance as a measurement.

The algorithm works as follows

1. First, we initialize k points, called means, randomly.
2. We categorize each item to its closest mean and we update the mean's coordinates, which are the averages of the items categorized in that mean so far.
3. We repeat the process until the algorithm converges that is the change in location of means after each iteration becomes very less or until some fixed number of iterations are done.

- **Gaussian Mixture model** - A Gaussian mixture model is a probabilistic model that assumes all the data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters. One can think of mixture models as generalizing k-means clustering to incorporate information about the covariance structure of the data as well as the centres of the latent Gaussians.

  Given a Gaussian mixture model, the goal is to maximize the likelihood function with respect to the parameters comprising the means and covariances of the components and the mixing coefficients using the Expectation-Maximization (EM) algorithm

  The algorithm is as follows

- Initialize the means, covariances and mixing coefficients, and evaluate the initial value of the log-likelihood.
- E step - Evaluate the responsibilities using the current parameter values
- M step. Re-estimate the parameters using the current responsibilities

- Evaluate log-likelihood
- If there is no convergence, return to step 2

Experiments

KMeans -

We have used the following Internal cluster validation Techniques for KMeans algorithm to find the optimum number of clusters

Elbow method	Silhouette Coefficient
Davies-Bouldin Index	Calinski-Harabasz Index

1. **Elbow method** - In this method, we plot a graph of Within-Cluster-Sum-of-Squares (WCSS) with the number of clusters K. The plot shows an 'elbow' which demarks a significant drop in the rate of decrease of WCSS which means after this point there is not much improvement in the model by increasing number of clusters. Selecting the number of clusters corresponding to the elbow point is the optimum number of clusters

2. **Silhouette Coefficient** - The Silhouette Coefficient is defined for each sample and is composed of two scores -

   a: The mean distance between a sample and all other points in the same class.

   b: The mean distance between a sample and all other points in the next nearest cluster.

   The Silhouette Coefficient s for a single sample is then given as:

$$S = \frac{b-a}{\max(a,\ b)}$$

   A higher Silhouette Coefficient score relates to a model with better defined clusters

3. **Davies-Bouldin Index** - This index signifies the average 'similarity' between clusters, where the similarity is a measure that compares the distance between clusters with the size of the clusters themselves. A lower Davies-Bouldin index relates to a model with better separation between the clusters. Zero is the lowest possible score. Values closer to zero indicate a better partition.

4. **Calinski-Harabasz Index** - The index is the ratio of the sum of between-clusters dispersion and of inter-cluster dispersion for all clusters (where dispersion is defined as the sum of distances squared) It is also known as the Variance Ratio Criterion. A higher Calinski-Harabasz score relates to a model with better defined clusters

Since we have labelled class data for the Number of clusters = 2. We have used the following **External cluster validation** Techniques for KMeans algorithm for K=2

Accuracy, Precision, Recall and f1 score	Adjusted Rand index
Normalized Mutual Information	homogeneity_score
completeness_score	v_measure_score
Fowlkes-Mallows scores	Contingency Matrix

1. **Accuracy, Precision, Recall and f1 score** – we have calculated these using predicted_labels (Labels we got after using some functions on predicted_model_labels). predicted_model_labels are labels that are directly predicted by the model.

2. **Adjusted Rand index** - Given the knowledge of the ground truth class assignments labels_true and our clustering algorithm assignments of the same samples labels_pred, the adjusted Rand index is a function that measures the similarity of the two assignments, ignoring permutations and with chance normalization. Perfect labelling has score 1.0, Bad (e.g. independent labellings) have negative or close to 0.0 scores.

3. **Normalized Mutual Information** - Given the knowledge of the ground truth class assignments labels_true and our clustering algorithm assignments of the same samples labels_pred, the Normalized Mutual Information is a function that measures the agreement of the two assignments, ignoring permutations. Perfect labelling has a score of 1.0. Bad (e.g. independent labellings) have non-positive scores.

4. **homogeneity_score** - This score measures desirable homogeneity objective which is for any cluster assignment defined as below
   homogeneity - each cluster contains only members of a single class. It is bounded below by 0.0 and above by 1.0 (higher is better).

5. **completeness_score** - This score measures desirable completeness objective which is for any cluster assignment defined as below
   completeness - all members of a given class are assigned to the same cluster. It is bounded below by 0.0 and above by 1.0 (higher is better).

6. **v_measure_score -** It is a Harmonic mean of completeness_score and homogeneity_score. A score of 0.0 is as bad as it can be, 1.0 is a perfect score.

7. **Fowlkes-Mallows scores** - The Fowlkes-Mallows score FMI is defined as the geometric mean of the pairwise precision and recall.

$$FMI = \frac{TP}{\sqrt{(TP+FP)*(TP+FN)}}$$

   Where TP is the number of True Positive, FP is the number of False Positive and FN is the number of False Negatives

8. **Contingency Matrix** - It reports the intersection cardinality for every true/predicted cluster pair. The contingency matrix provides sufficient statistics for all clustering metrics where the samples are independent and identically distributed and one doesn't need to account for some instances not being clustered.

Gaussian Mixture Model -

We have used the same **External cluster validation** Techniques for **Gaussian Mixture Model**

We have used all the **Internal cluster validation** Techniques of KMeans Except Elbow method in **Gaussian Mixture Model** along with 2 additional validation Techniques –

1. **Akaike Information Criterion** (AIC) – It is an estimator of out-of-sample prediction error and thereby relative quality of statistical models for a given set of data. Given a collection of models for the data, AIC estimates the quality of each model, relative to each of the other models

   We plot AIC curve then we plot its gradient. let n be the number of clusters at which the gradient stops increasing significantly. The number of clusters one less than this i.e. n-1 is the optimum number of clusters.

2. **Bayesian Information Criterion** (BIC) – It is a criterion for model selection among a finite set of models. It is based, in part, on the likelihood function and it is closely related to the Akaike information criterion (AIC).

   We plot BIC curve then we plot its gradient let n be the number of clusters at which the gradient stops increasing significantly. The number of clusters one less than this i.e. n-1 is the optimum number of clusters.

Dataset description, Pre-processing and Data Analysis

We are using UCI ML repository breast cancer data this data has 11 columns which are as follows –

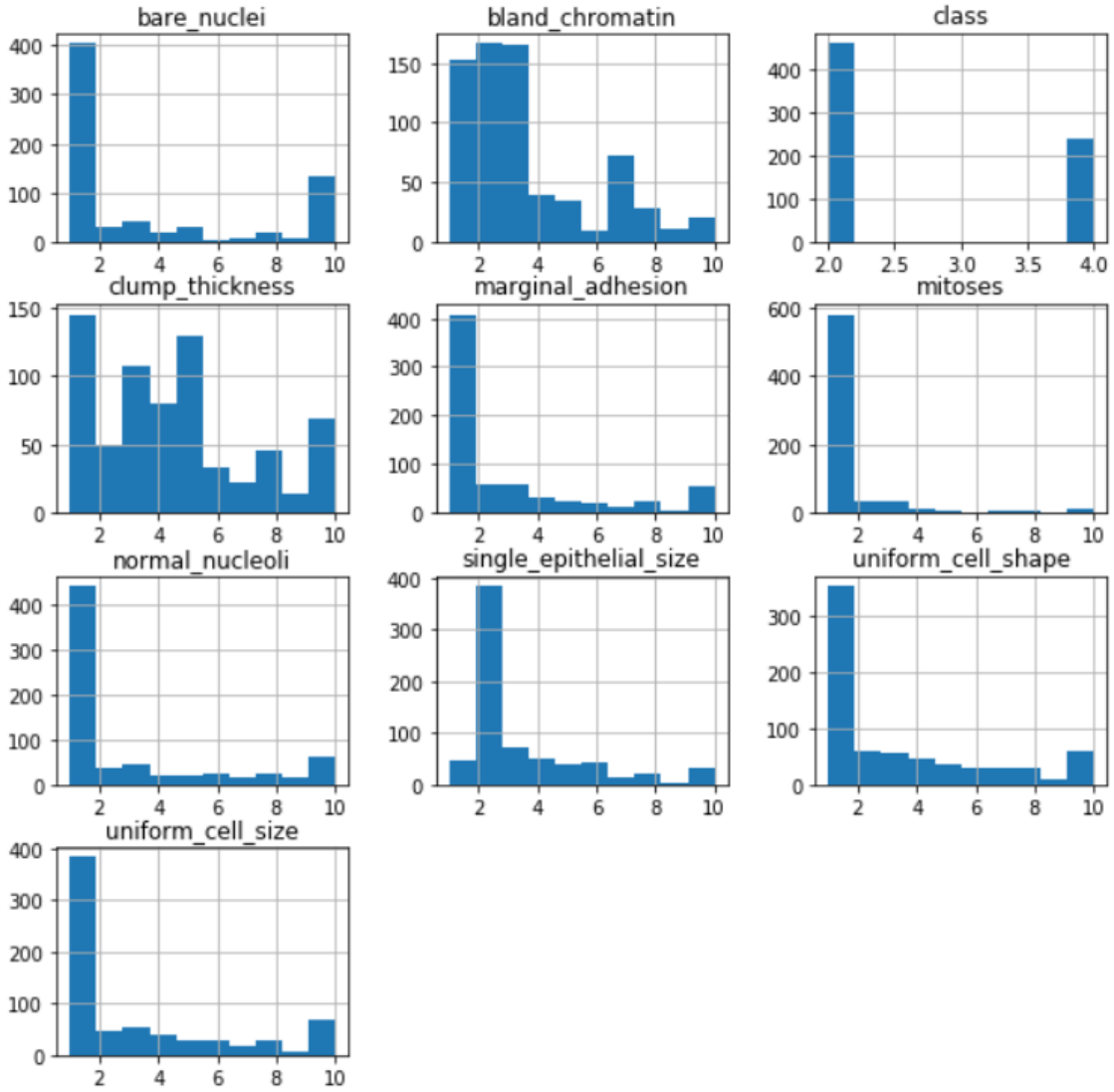
- id
- clump_thickness
- uniform_cell_size
- uniform_cell_shape
- marginal_adhesion
- single_epithelial_size
- bare_nuclei
- bland_chromatin'
- normal_nucleoli
- mitoses
- class

We dropped the 1st column which is id because it is of no use in ML models. This dataset also has missing values in the bare_neclei column and the values in this column are of str type so we first replaced all missing values of this column with the mean of this column and then converted all values of the column to float64 type.

Then we checked the description of the data set we observed that the min value of every feature except the class column is 1 and the maximum is 10. Also mean is very less for all features except class column.

mean of the class column is less than 3 it means we have more benign cases than malignant

We plotted a Histogram of each feature and observed that clump_thickness is very evenly distributed and bland_chromatin is left-skewed distributed. All other features are concentrated at one specific value with very less count for other values



Then we plotted scatter plot matrix of data set to find if there exist any features with a linear correlation between them, we found that uniform_cell_size and uniform_cell_shape have a high linear correlation. We verified it with a correlation matrix of data set which shows that these two features have a correlation of 0.906882 which is very high. All other features do not have a high correlation

Therefore, we made 3 cases –

Case A - Kept both parameters while applying unsupervised clustering algorithms

Case B - removed uniform_cell_size feature and kept uniform_cell_shape feature in the data set

Case C - removed uniform_cell_shape feature and kept uniform_cell_size parameter in the data set

GMM – For GMM Case C gave the best external cluster validation results and Internal cluster validation results remained the same in all 3 cases therefore we removed uniform_cell_shape from data set for GMM.

KMeans – For KMeans Case B gave the best external cluster validation results and Internal cluster validation results remained the same in all 3 cases, therefore, we removed uniform_cell_size from data set for KMeans.

We also dropped the Class column when we created the X matrix to fit into both of the models because it is the unsupervised analysis, we can't put class labels inside models. Although we created a y array from class labels which we used for external cluster analysis.

Then we applied feature scaling to the X matrix and standardized the data so that each feature column has 0 mean and unity variance.

Experimental Setup

We have used Jupyter Notebook to write code using python3 programming language.

Following Libraries were used to conduct experiments –

- NumPy version - 1.16.5
- pandas version - 0.25.1
- Matplotlib version - 3.1.1
- scikit-learn version - 0.21.3

Experimental Procedure

KMeans

Internal Cluster Validation

- generated Elbow graph by plotting WCSS with different numbers of clusters.
- Plotted Silhouette Coefficient with different number of clusters
- Plotted davies_bouldin_score with different number of clusters
- Plotted calinski_harabasz_score with different number of clusters

External Cluster Validation

Since we have the class labels for K=2 we can perform External Cluster analysis for K=2 where K is the number of clusters

We created two functions described below to calculate accuracy, precision, recall and F1 score

- infer_cluster_labels – This function takes predicted_model_labels and actual class_labels i.e. y and associates most probable label with each cluster number in the model and returns a dictionary of clusters number assigned to each class label
- infer_data_labels – This function takes dictionary returned by the above function and predicted_model_labels. It determines class label for each element in

predicted_model_labels depending on the cluster it has been assigned to and returns predicted_labels.

- calculated accuracy, precision, recall and f1 score from predicted_labels and y using above 2 functions.
- Calculated Adjusted Rand index using y and predicted_model_labels
- Calculated Normalized Mutual Information using y and predicted_model_labels
- Calculated homogeneity_score using y and predicted_model_labels
- Calculated completeness_score using y and predicted_model_labels
- Calculated v_measure_score using y and predicted_model_labels
- Calculated Fowlkes-Mallows scores using y and predicted_model_labels
- Calculated Contingency Matrix using y and predicted_model_labels.

Gaussian Mixture Model

External Clustering Validation - Same as of KMeans above.

Internal Clustering Validation – Same as of KMeans above excluding Elbow Method and with 2 more validation techniques.

1. Generated the AIC curve by plotting the AIC score with different number of clusters. plotted its gradient with different number of clusters, let n be the number of clusters at which the gradient stops increasing significantly. The number of clusters one less than this i.e. n-1 is the optimum number of clusters.

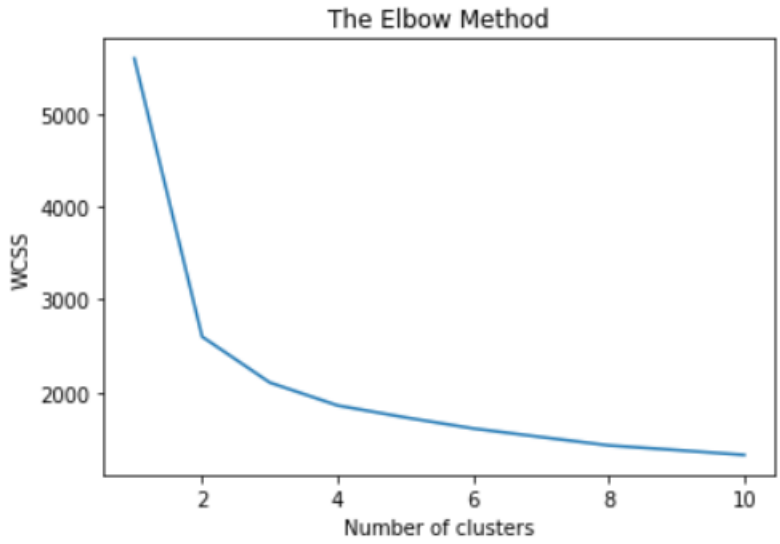
2. Generated the BIC curve by plotting the BIC score with different number of clusters. plotted its gradient with different number of clusters, let n be the number of clusters at which the gradient stops increasing significantly. The number of clusters one less than this i.e. n-1 is the optimum number of clusters.

Results observed

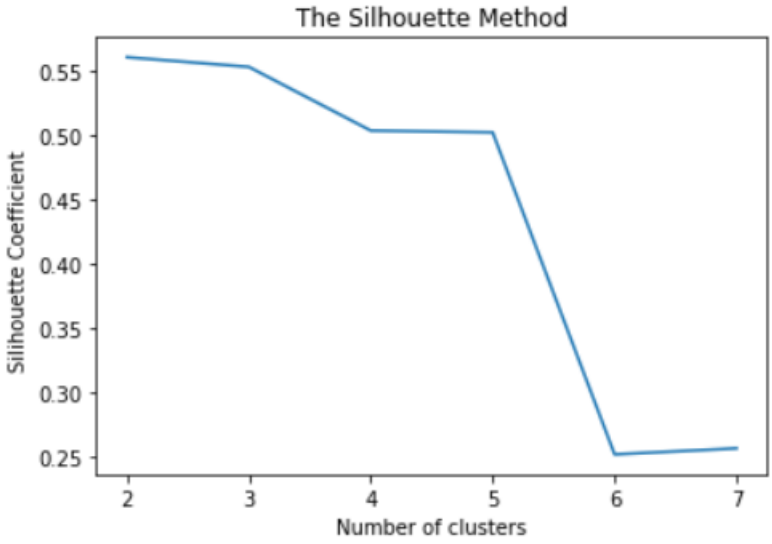
K Means

Internal Cluster Validation

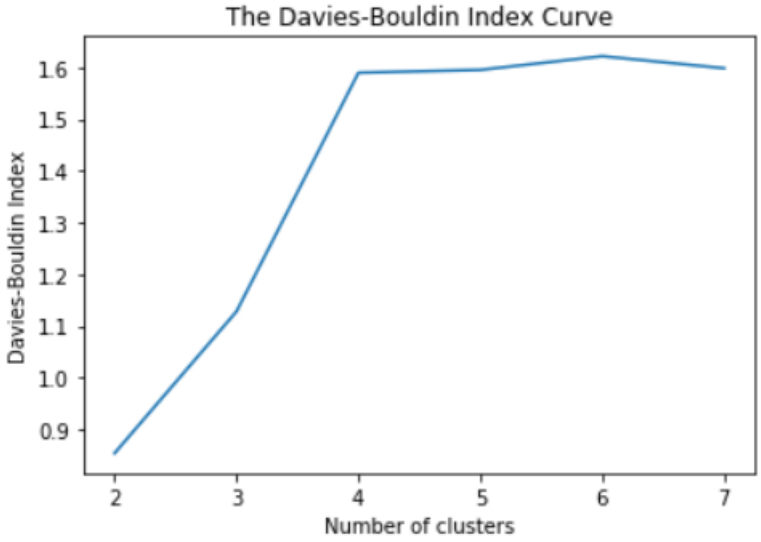
1. **Elbow method** – We found elbow of the graph at K=2 so the optimum number of clusters according to this metric is 2.



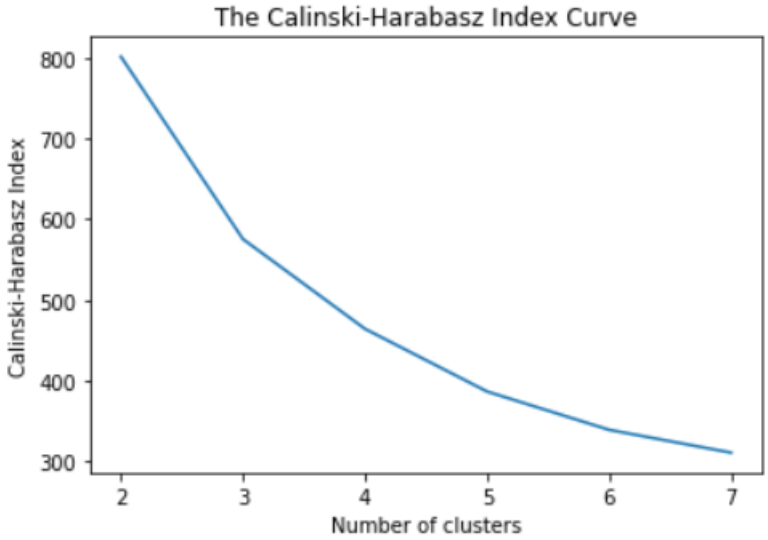
2. **Silhouette Method** – We found the maximum value of the silhouette index at K=2. So, the optimum number of clusters according to this metric is 2.



3. **Davies-Bouldin Index Curve** – We found the minimum value of the Davies-Bouldin Index at K=2. So, the optimum number of clusters according to this metric is 2.



4. **Calinski-Harabasz Index Curve**- We found the maximum value of the Calinski-Harabasz Index at K=2. So, the optimum number of clusters according to this metric is 2.



From the above 4 metrics, we got the same results that is the optimum number of clusters = 2. So, we conclude the optimum number of clusters is indeed 2.

External Clustering Validation (for K=2)

1. **Accuracy, Precision, Recall and f1 score** – The results are as follows –

   Accuracy – 0.9642346208869814

Class	precision	recall	F1-Score
2	0.97	0.97	0.97
4	0.95	0.95	0.95

2. **Adjusted Rand index** – We got the Adjusted Rand index = 0.860620941314958.

3. **Normalized Mutual Information** – We got Normalized Mutual Information = 0.7636527875368617

4. **homogeneity_score** - We got homogeneity_score = 0.7641945853723171

5. **completeness_score** - We got completeness_score = 0.7631117574035869

6. **v_measure_score** - We got v_measure_score = 0.7636527875368617

7. **Fowlkes-Mallows scores** - We got Fowlkes-Mallows scores = 0.9368751428351904

8. **Contingency Matrix** – the contingency matrix is shown below

13	445
229	12

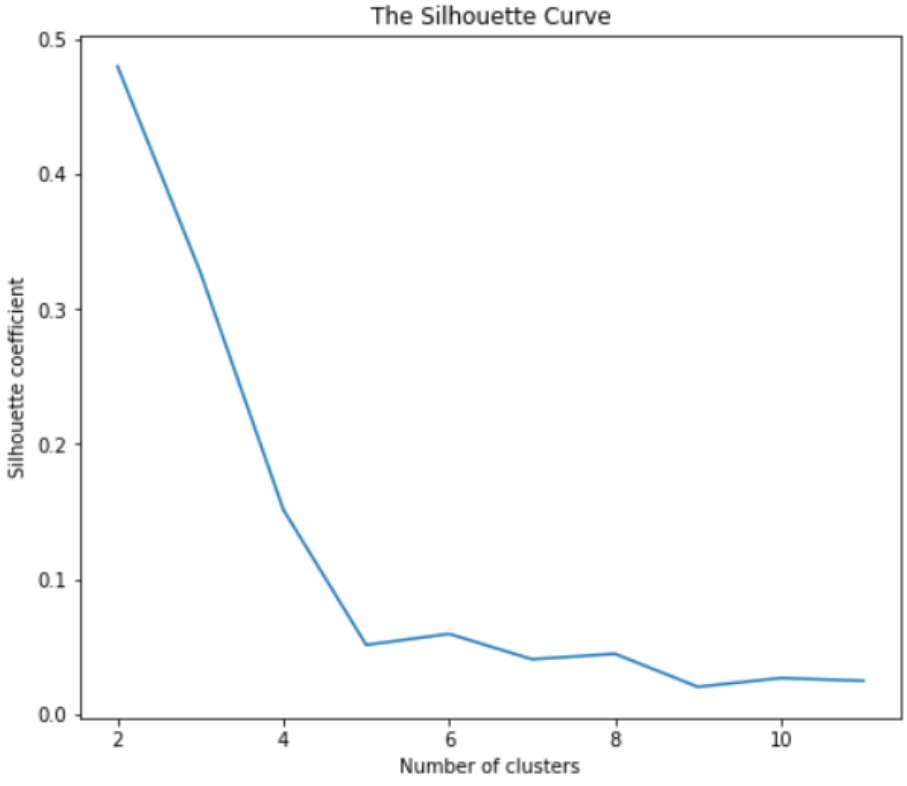
From above we can conclude there are 458 samples whose true cluster is 2 Of them, 13 are in predicted cluster 0 and 445 are in predicted cluster 1

we can also conclude that there are 241 samples whose true cluster is 4 Of them, 229 are in predicted cluster 0 and 12 are in predicted cluster 1

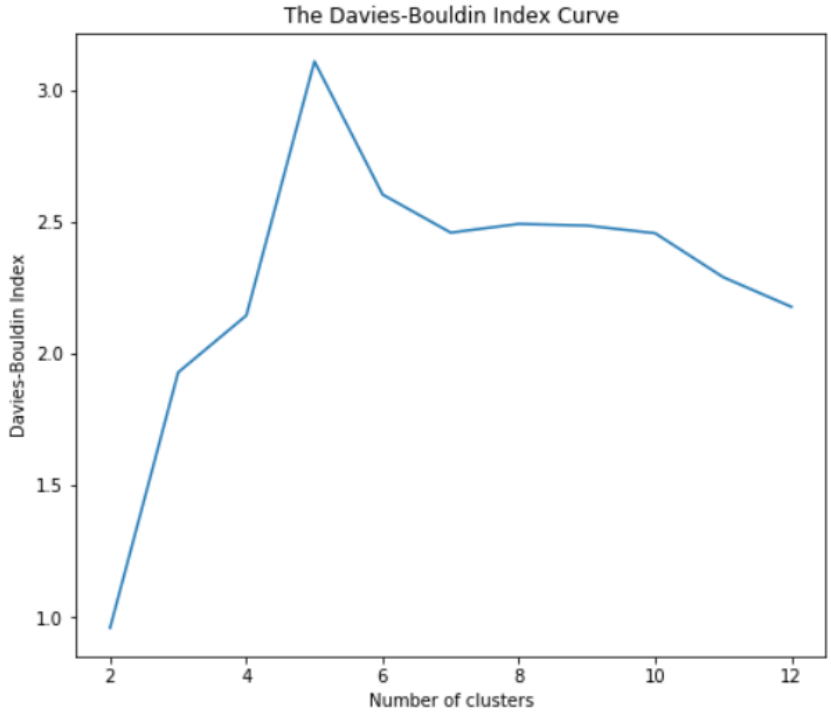
Gaussian Mixture Model

Internal Cluster Validation

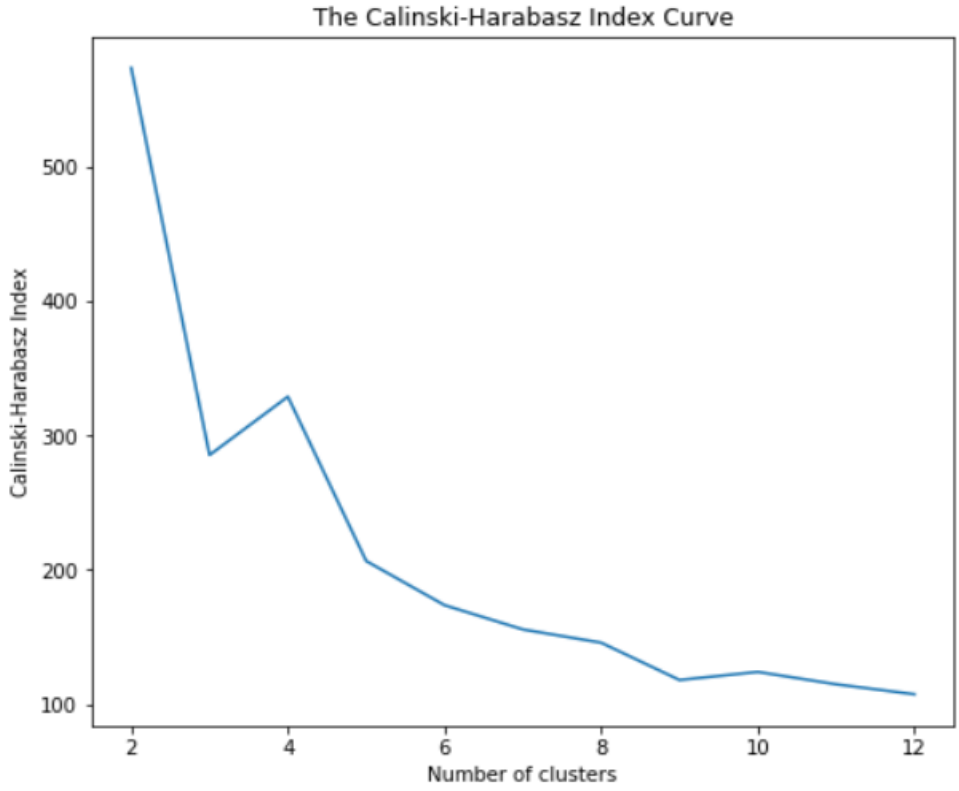
1. **Silhouette Method** – We found the maximum value of the silhouette index at K=2. So, the optimum number of clusters according to this metric is 2.



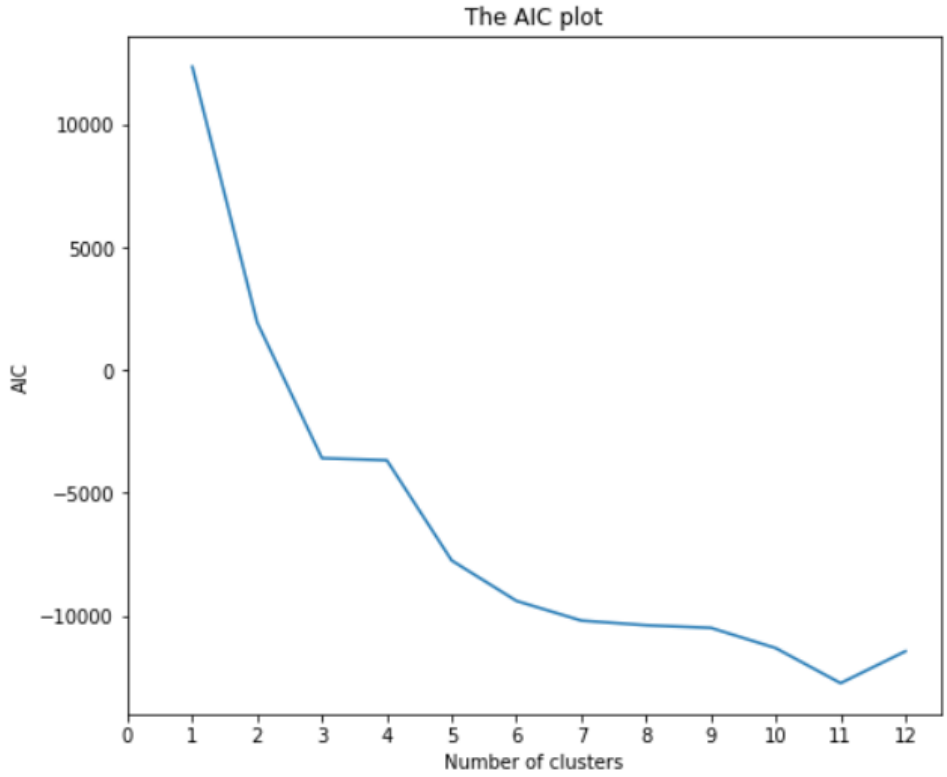
2. **Davies-Bouldin Index Curve** – We found the minimum value of the Davies-Bouldin Index at K=2. So, the optimum number of clusters according to this metric is 2.



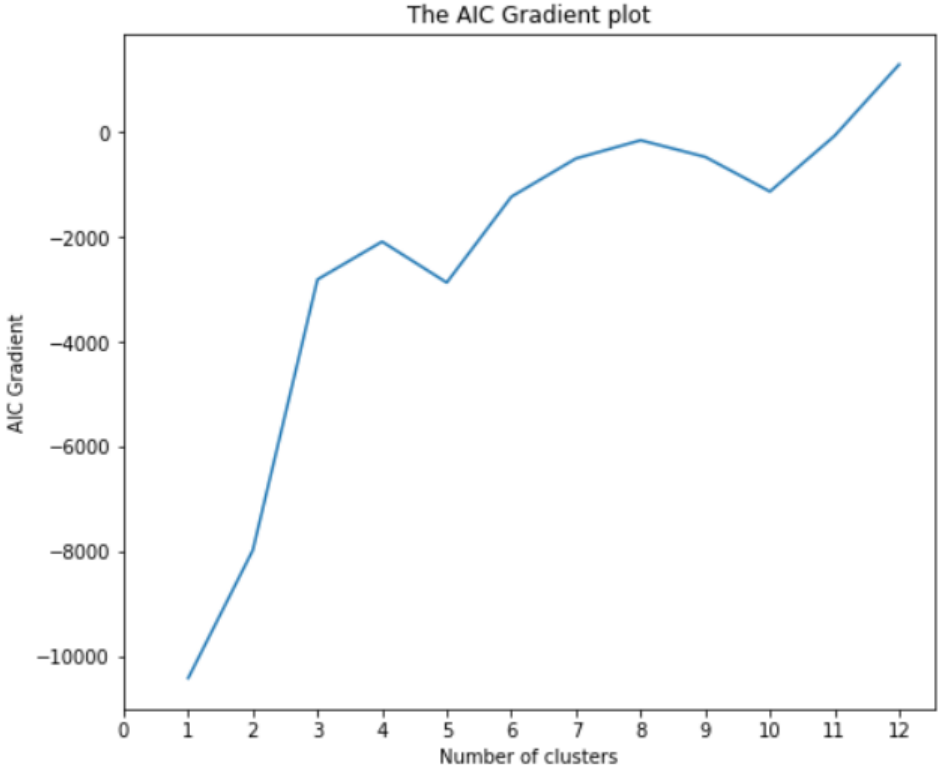
3. **Calinski-Harabasz Index Curve**- We found the maximum value of the Calinski-Harabasz Index at K=2. So, the optimum number of clusters according to this metric is 2.



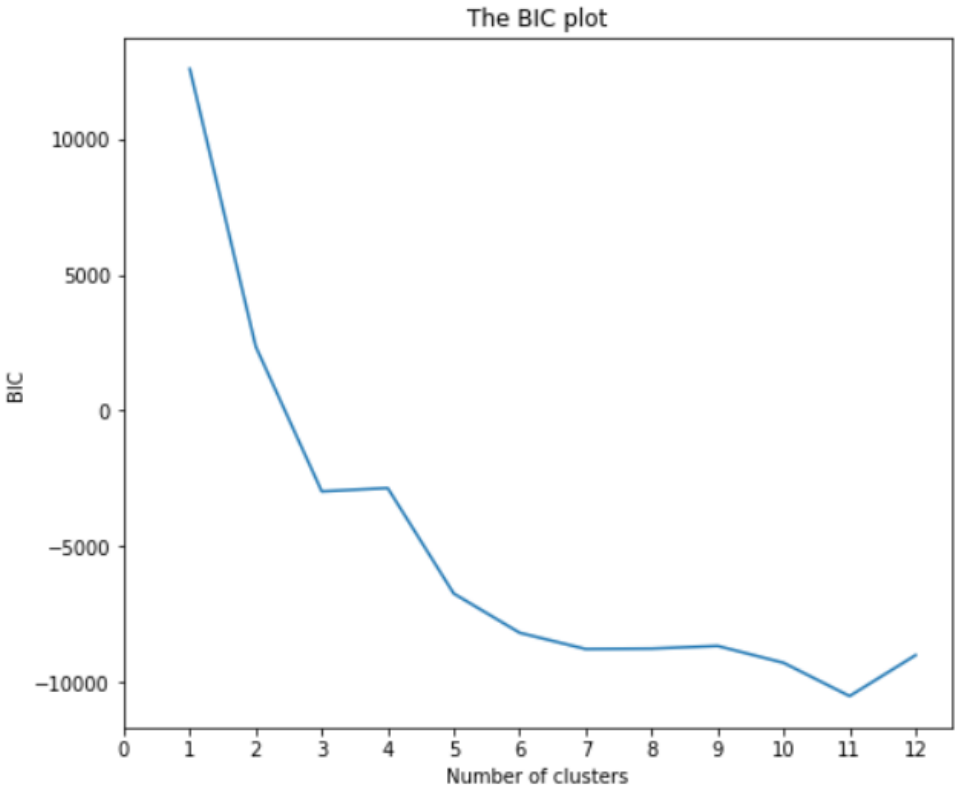
4. **Akaike Information Criterion** – Below is the AIC curve. alone it is not of much use but its gradient is used to calculate optimum number of clusters



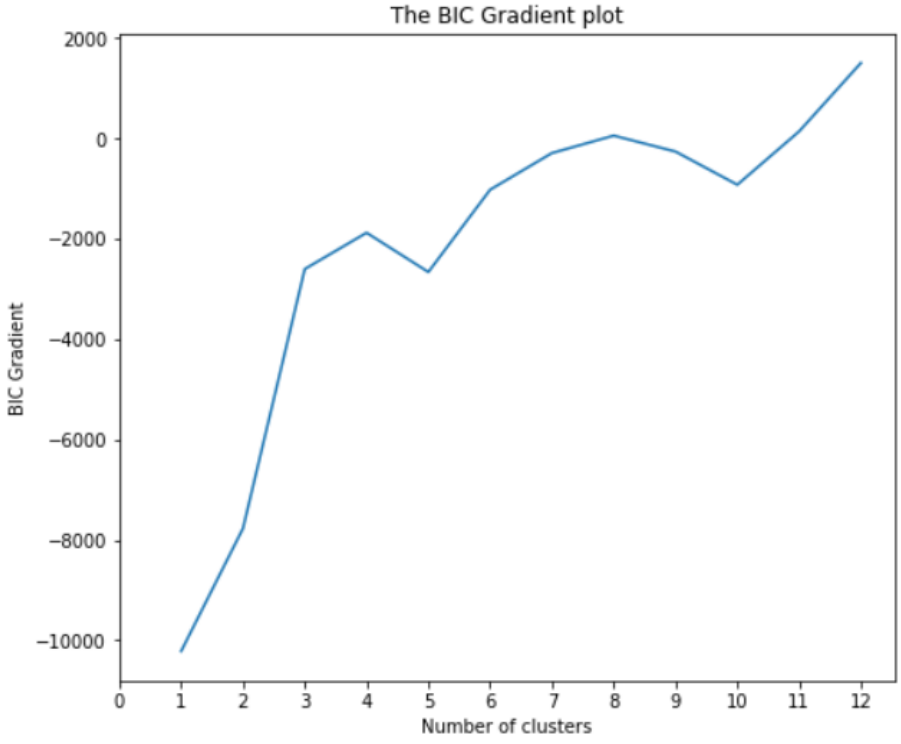
Gradient of AIC curve – following is gradient of AIC curve plotted with different number of clusters. As we can observe its gradient stops increasing significantly (becomes almost constant) at K=3 so optimum number of clusters is therefore K=2



5. **Bayesian Information Criterion** – Below is the BIC curve. alone it is not of much use but its gradient is used to calculate optimum number of clusters



Gradient of BIC curve – following is gradient of BIC curve plotted with different number of clusters. As we can observe its gradient stops increasing significantly (becomes almost constant) at K=3 so optimum number of clusters is therefore K=2



From the above 5 metrics, we got the same results that is the optimum number of clusters = 2. So, we conclude the optimum number of clusters is indeed 2.

External Clustering Validation (for K=2)

1. **Accuracy, Precision, Recall and f1 score** – The results are as follows –

   Accuracy – 0.9256080114449213

Class	precision	recall	F1-Score
2	1	0.89	0.94
4	0.82	1	0.90

2. **Adjusted Rand index** – We got the Adjusted Rand index = 0.7235238012982509

3. **Normalized Mutual Information** – We got Normalized Mutual Information = 0.6769101244558279

4. **homogeneity_score** – We got homogeneity_score = 0.6957564343841053

5. **completeness_score** - We got completeness_score = 0.6590578868306297

6. **v_measure_score** – We got v_measure_score = 0.6769101244558279

7. **Fowlkes-Mallows scores** - We got Fowlkes-Mallows scores = 0.870361666211552

8. **Contingency Matrix** – the contingency matrix is shown below

52	406
241	0

From above we can conclude there are 458 samples whose true cluster is 2 Of them, 52 are in predicted cluster 0 and 406 are in predicted cluster 1

we can also conclude that there are 241 samples whose true cluster is 4 Of them, 241 are in predicted cluster 0 and 0 are in predicted cluster 1

Conclusion

By comparing External Clustering Validation results of Gaussian Mixture Model and KMeans we can conclude that KMeans performed better on given data set. It may be because as we have seen that features of dataset are not normally distributed and almost all features are concentrated on a value which makes it difficult for GMM to represent it as a mixture of Gaussian Curves