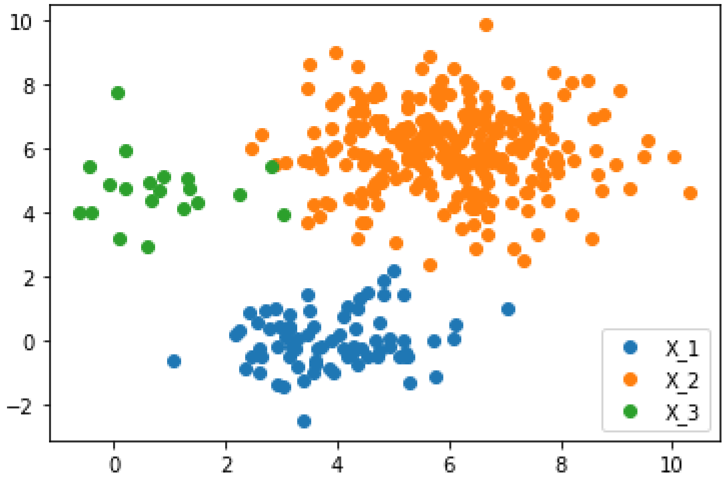
Vipin Kumar Seth
111701030

Generation of 3 clusters of different size and shape

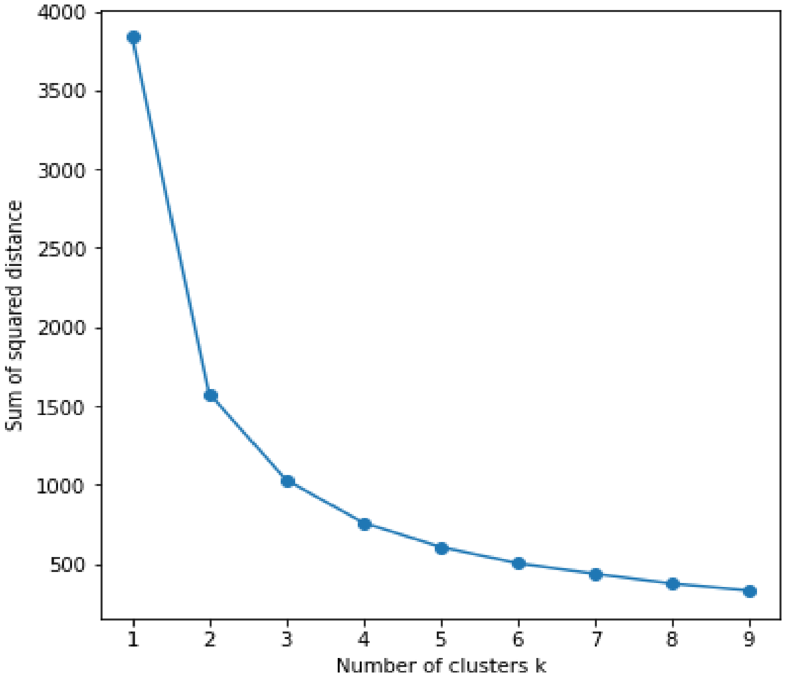
1. X_1 = np.random.multivariate_normal(mean=[4, 0], cov=[[1, 0], [0, 1]], size=75)
2. X_2 = np.random.multivariate_normal(mean=[6, 6], cov=[[2, 0], [0, 2]], size=250)
3. X_3 = np.random.multivariate_normal(mean=[1, 5], cov=[[1, 0], [0, 2]], size=20)

Plot of Data



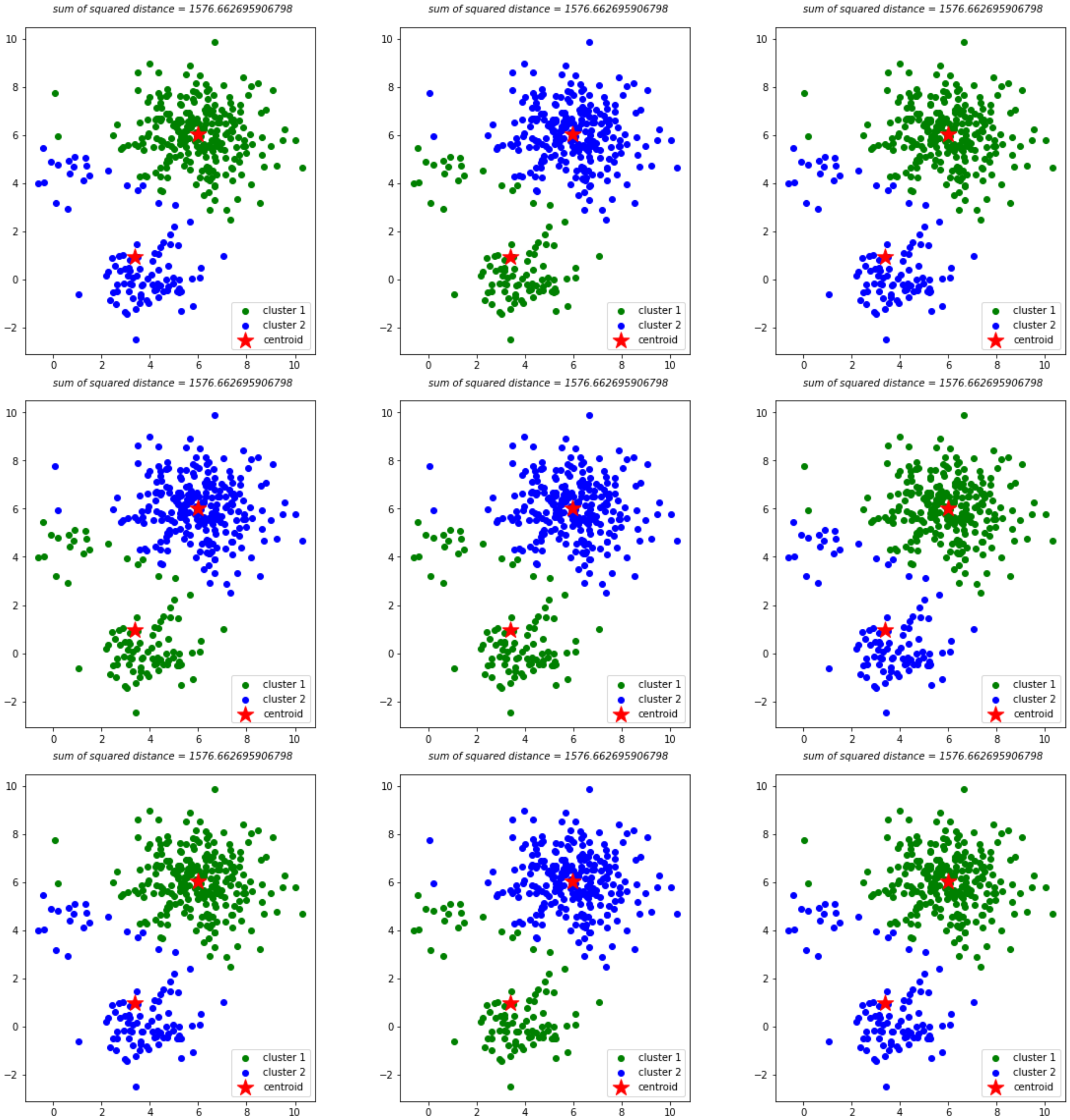
K-Means

To get the value of k , I used the elbow method(k = 1 to 9)
Sum of squared distance = [3831.67, 1541.63, 1021.28, 775.06, 608.30, 531.32, 468.14, 408.80, 353.17]

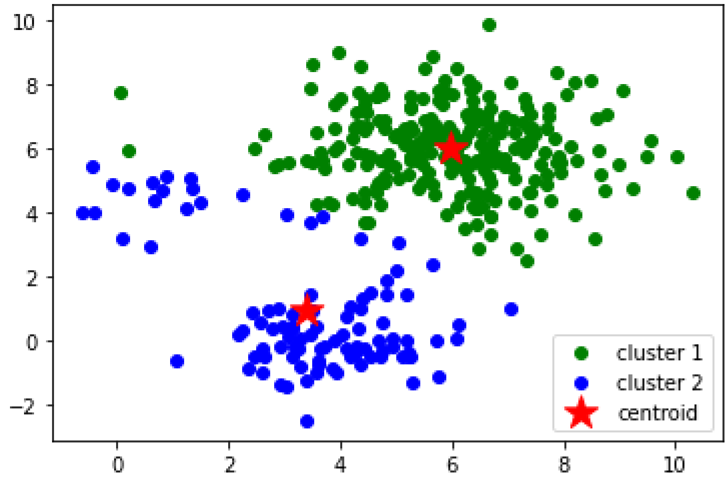


So here we can see that after k = 2, there is a jump in the values of the sum of squared distances(SSD). This is also evident from the values of ssd which is printed above.

So now we have the value of k = 2. Now performing the k-means algorithms for different initialization points.



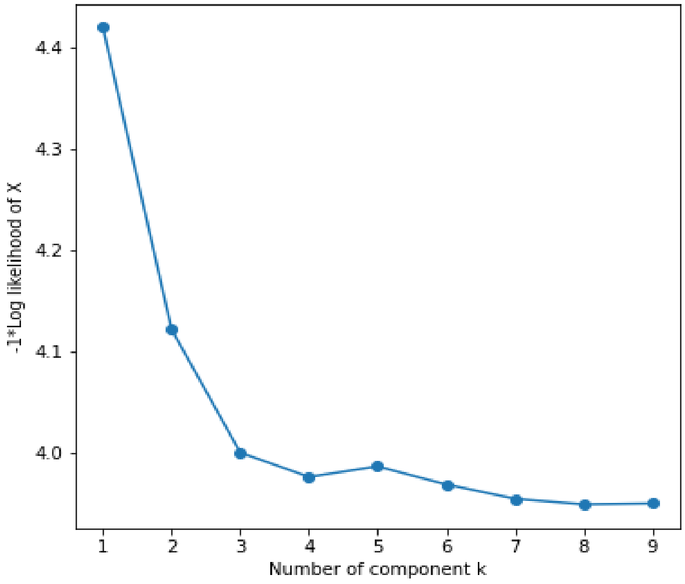
Now for all the different initialization is keeping track of the minimum SSD.
So the minimum squared distance is 1576.662695906798
Hence the final clusters is



Centre = [[5.98034785 6.02819829]
              [3.39348493 0.957126]]

Gaussian Mixture Model

To get the optimal number of components we can use the elbow method , here we will calculate log likelihood of the data.Same as the K-Means we will find a sharp turn in the log likelihood value.Below is the plot (here i plotted (-1)*log value in order to easily see the elbow point as log values were negatives)



So here we can see that at k = 3 , we can see an elbow point , indicating the optimal number of clusters.

Also we can use the **Akaike Information Criterion (AIC)** or the **Bayesian Information Criterion (BIC)** to further support the number of components. Let L be the maximum value of the likelihood function for the model, p be the number of estimated parameters in the model and N be the total number of data points.

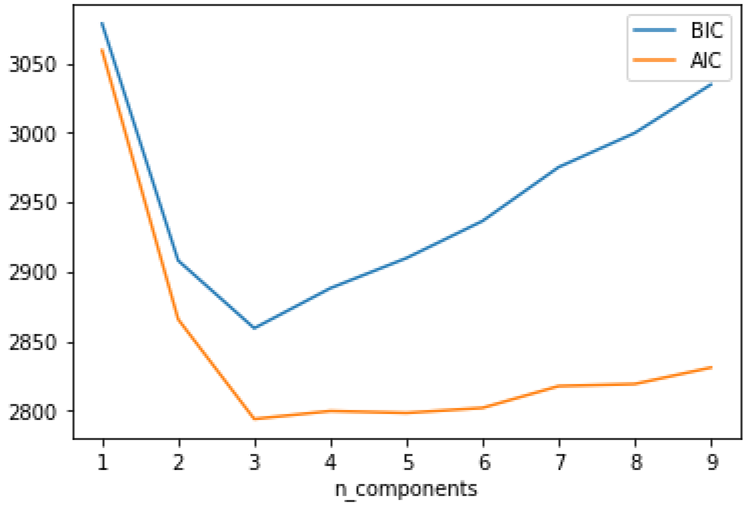
Then the AIC value of the model is the following:

$$\mathrm{AIC}=2\cdot p-2\cdot \ln(L)$$

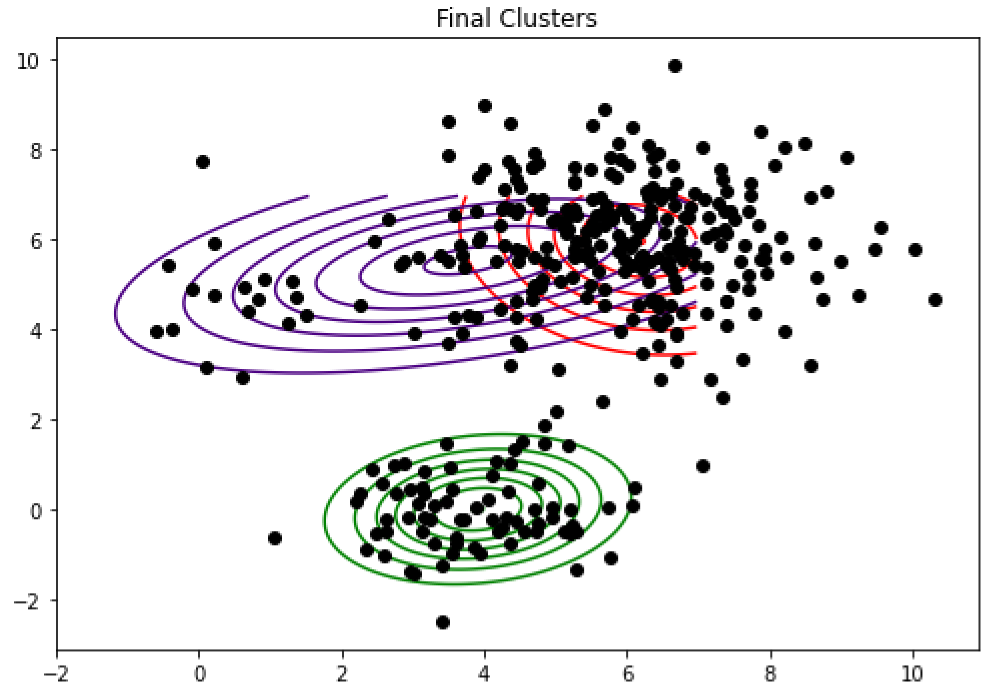
And the BIC value is denoted as:

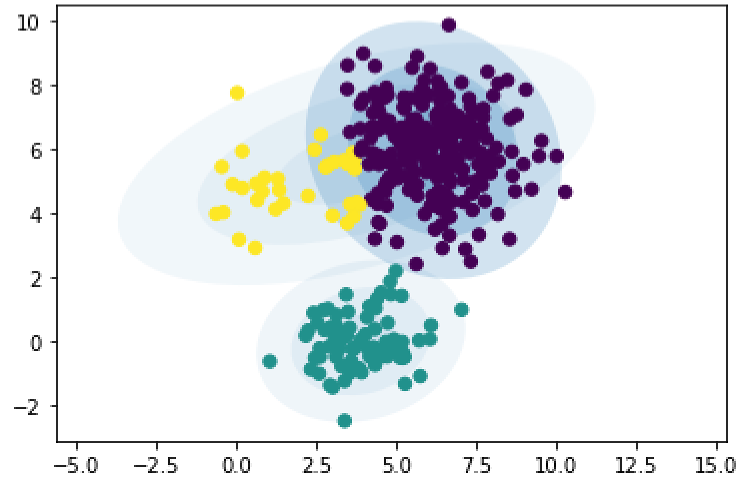
$$\mathrm{BIC}=-2\cdot \ln(L)+p\cdot \ln(N)$$

For both evaluation criteria, the lower the better.



So seeing the plot we can very easily verify that k = 3 is the optimal number of components.





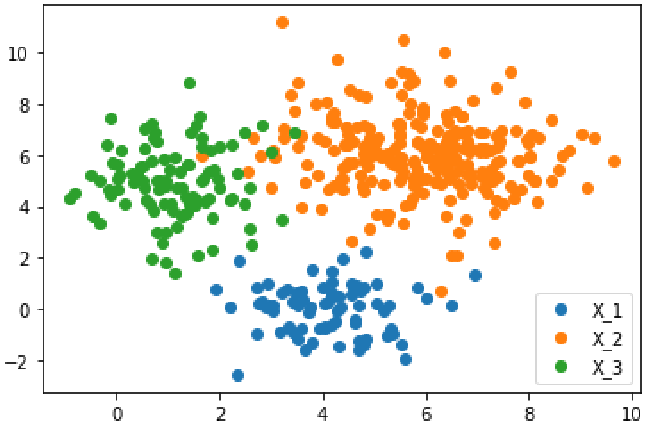
```
Mean =  [[3.91269467 0.01306824]
         [6.18157806 5.97155361]
        [3.77421788 5.53972595]]
```

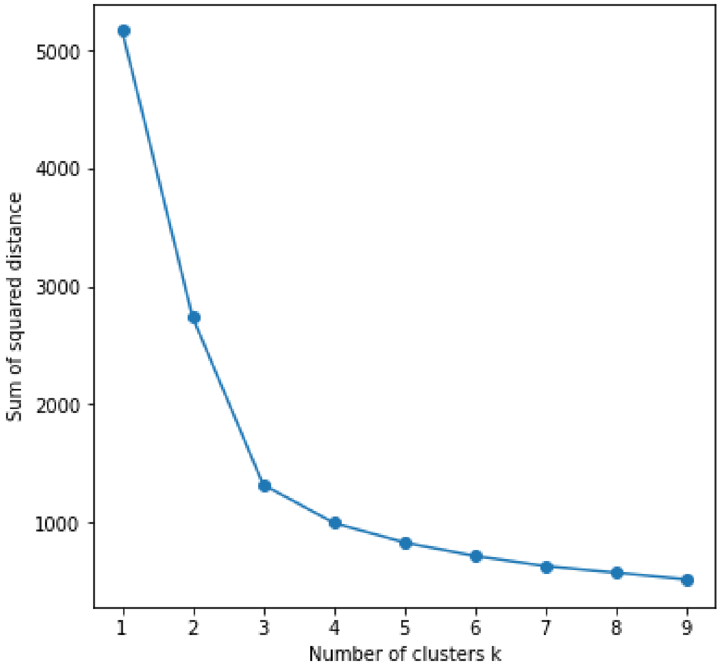
Comparison of K-Means and GMM on the basis of means of cluster:

In K-Means we found only two clusters even though we have 3 different clusters originally , this is due to the fact that the number of samples in X_3 is quite low . It is just

20 data samples.further X_3 is close to X_2. Hence K-Means is unable to recognize the third cluster and the elbow method has given only 2 clusters.

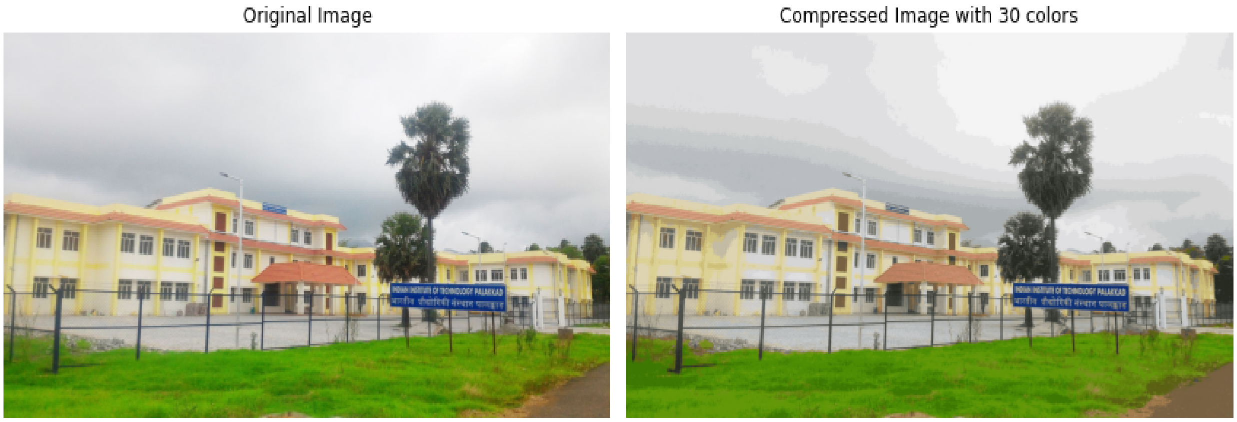
To further support my point , if there were more data samples in X_3 (let's say 100), then the plot will be





Now here we can see that k = 3 is the optimal k.

But for GMM even though the cluster samples were less in X_3 , it was able to find 3 clusters(although the mean is shifted a bit).
So here we can say that GMM has formed better than K-Means (Here we can claim this only because we know the original number of data. For real time data , we won’t be able to say which is better than. That will be completely dependent on the data and our requirement of number of clusters).



Original Image

Compressed Image with 30 colors

INDIAN INSTITUTE OF TECHNOLOGY PALAKKAD
INDIAN INSTITUTE OF TECHNOLOGY PALAKKAD

The image is compressed to have now 30 colors and the image size is the same as the original image, because we are just replacing each pixel value by it’s center value.