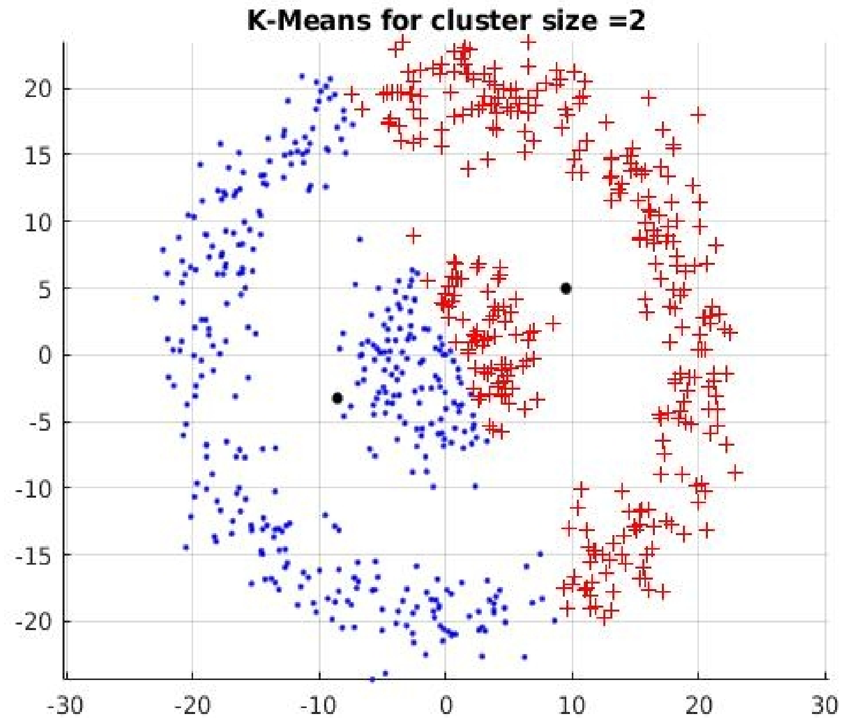
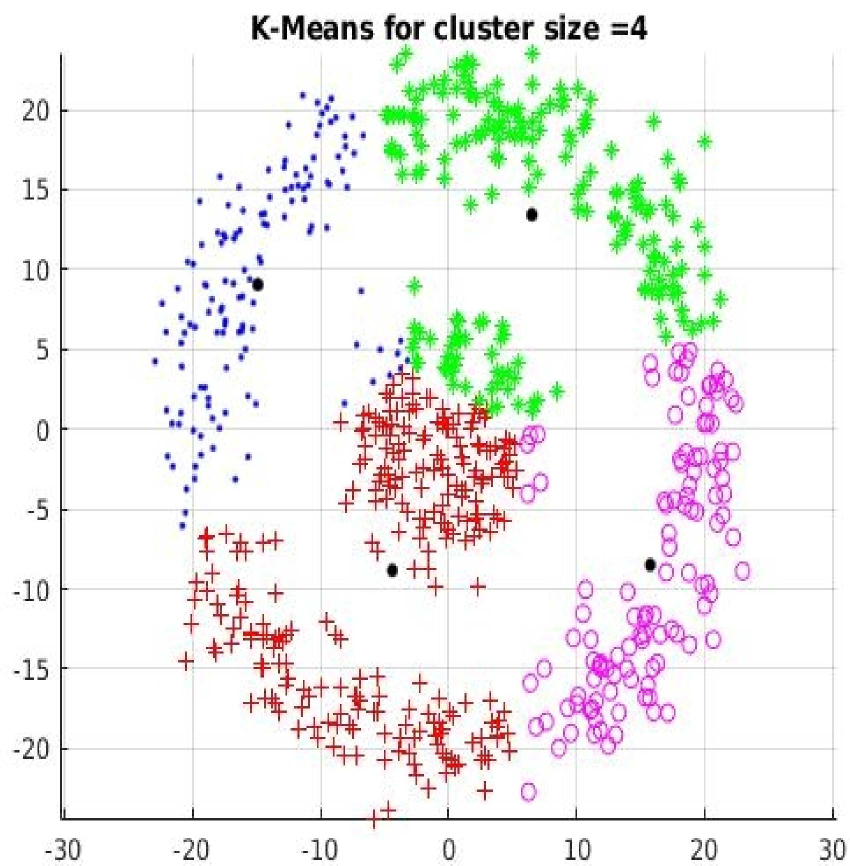
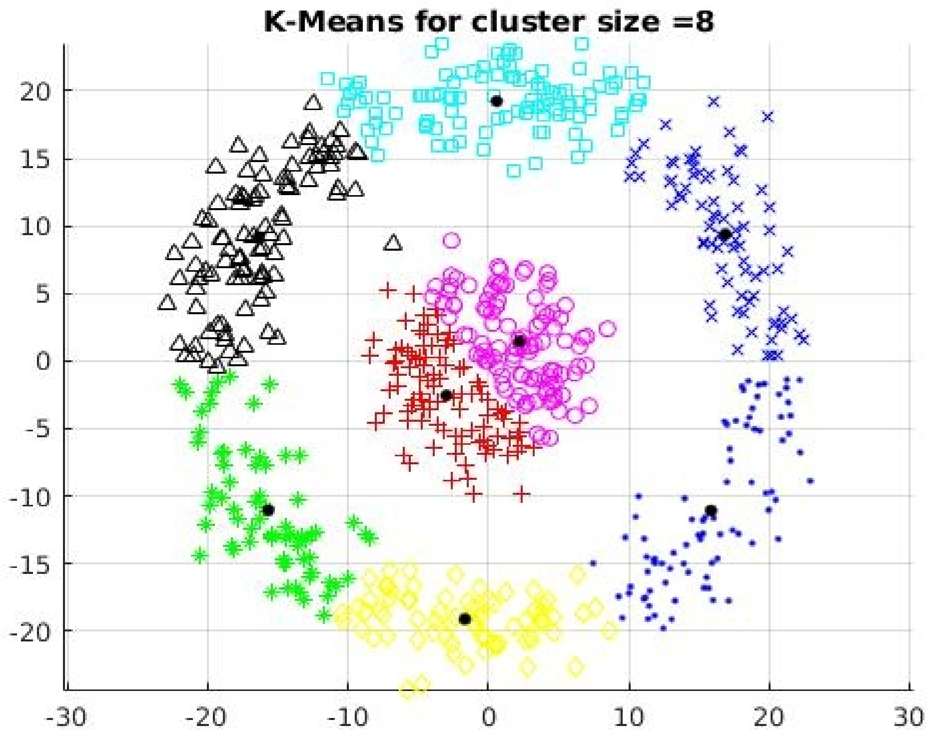
ENEE439M
Assignment 5

1. K-Means Clustering
Dataset 1 -

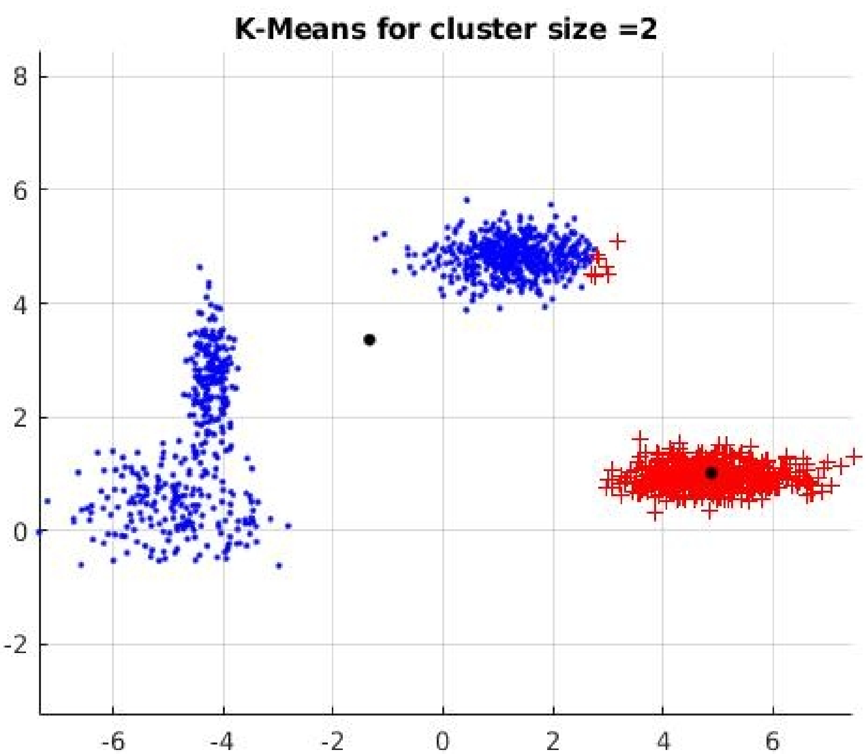




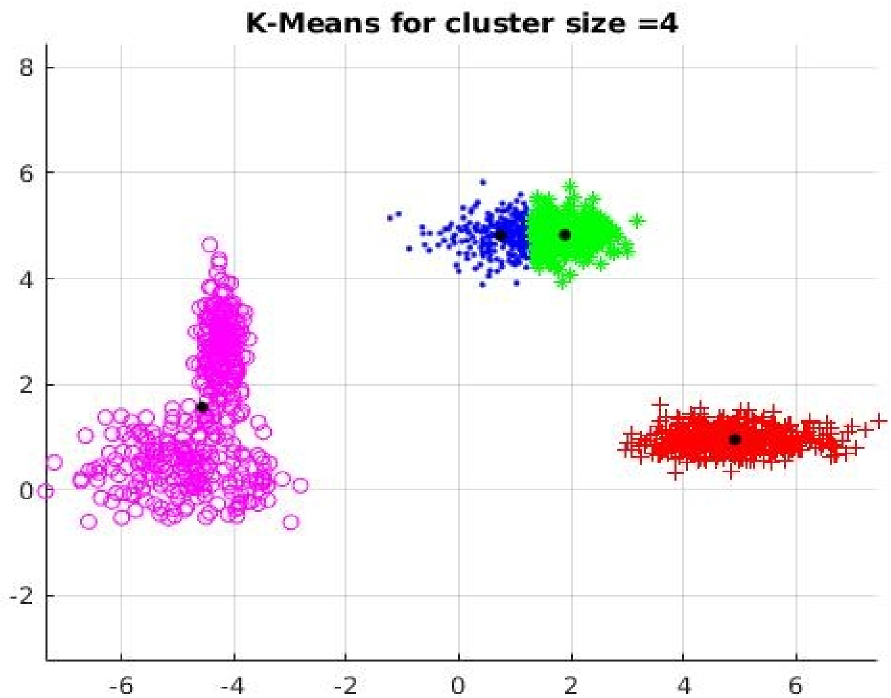
ENEE439M
Assignment 5

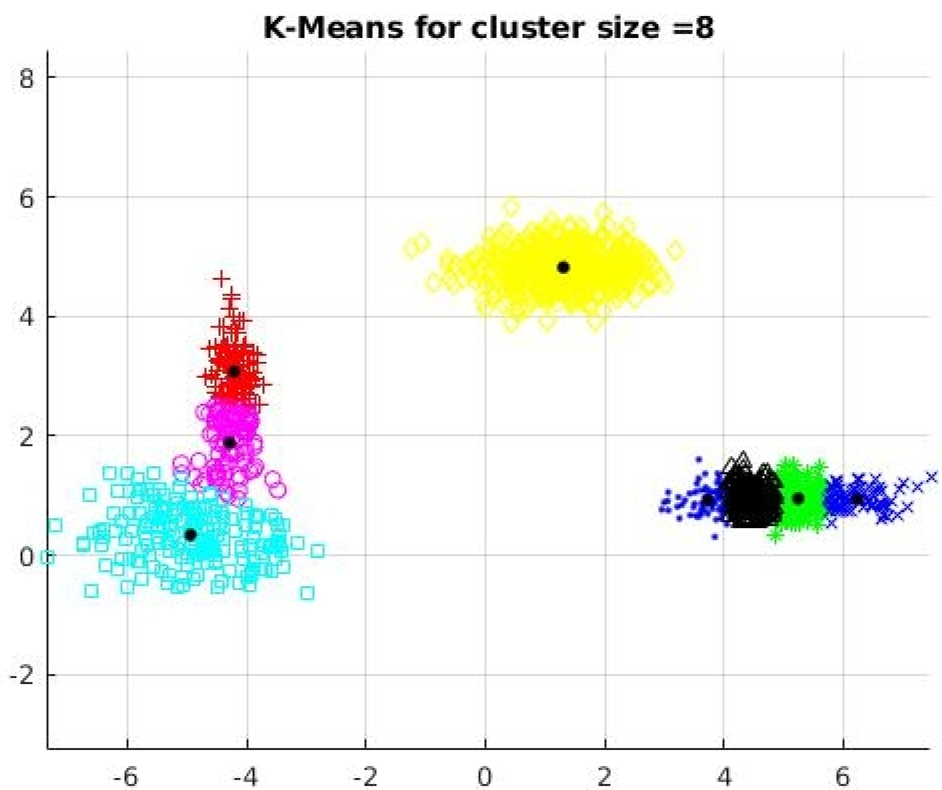


Dataset 2 -



ENEE439M
Assignment 5

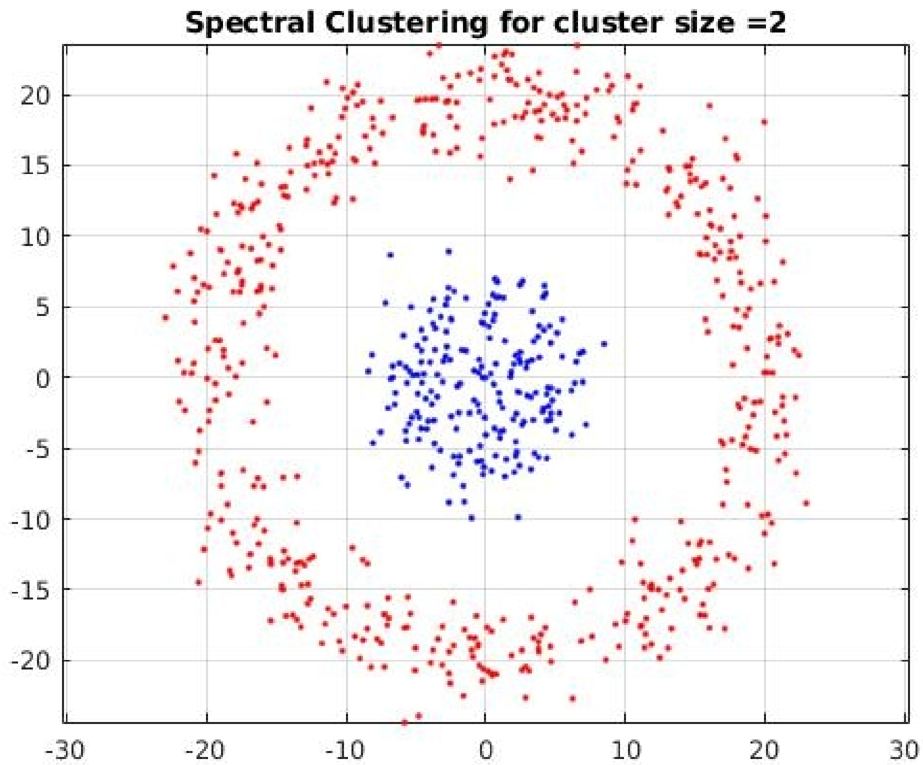


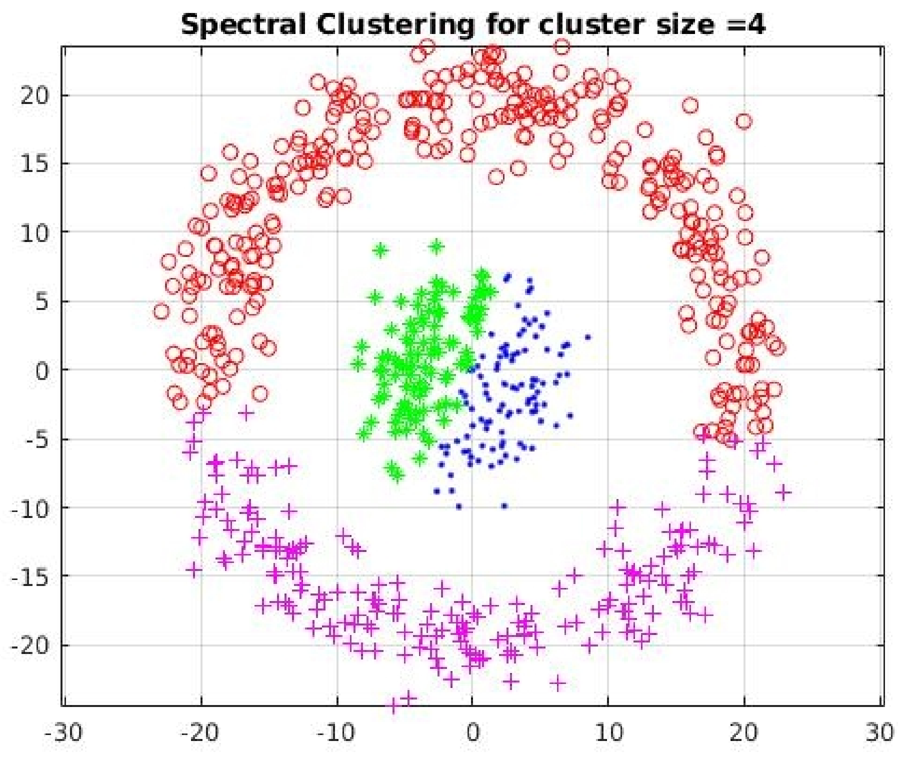


ENEE439M
Assignment 5

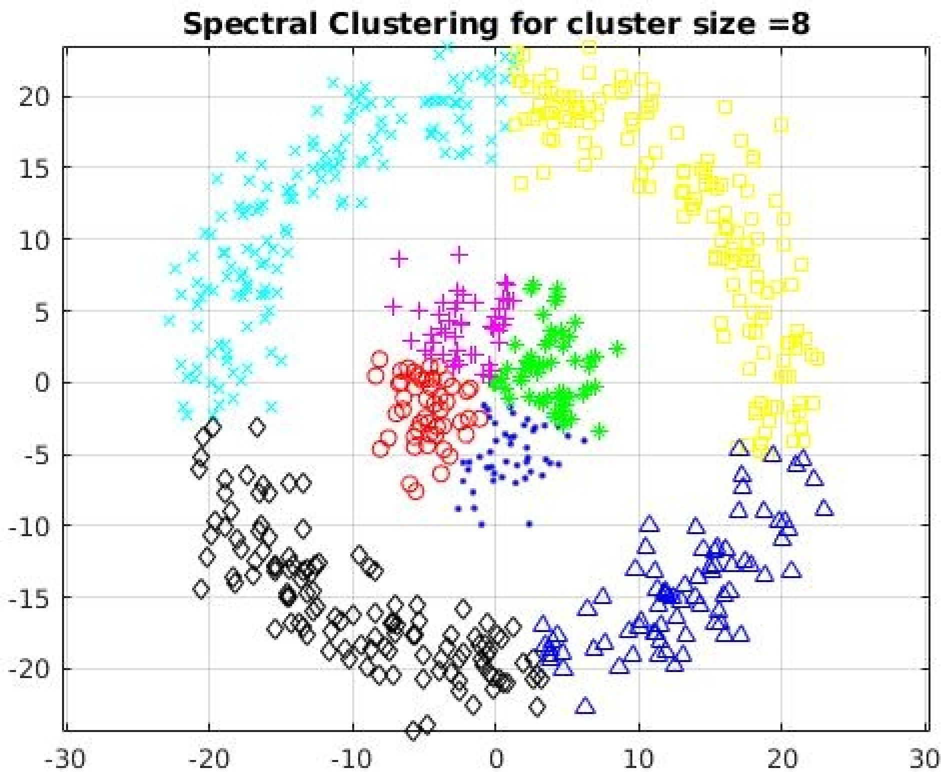
2. Spectral Clustering -

Dataset 1 -

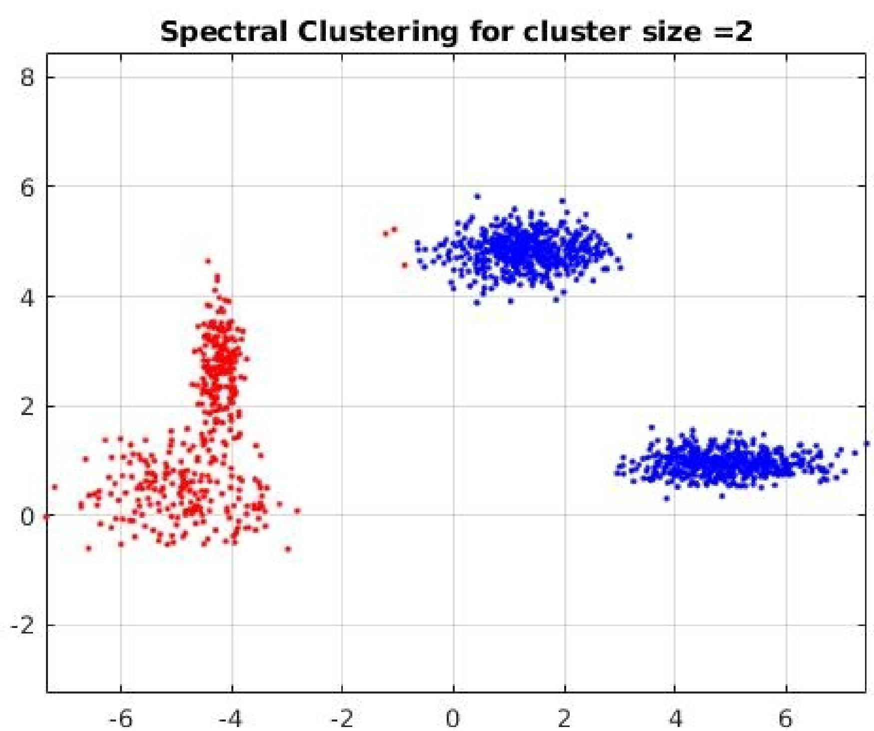




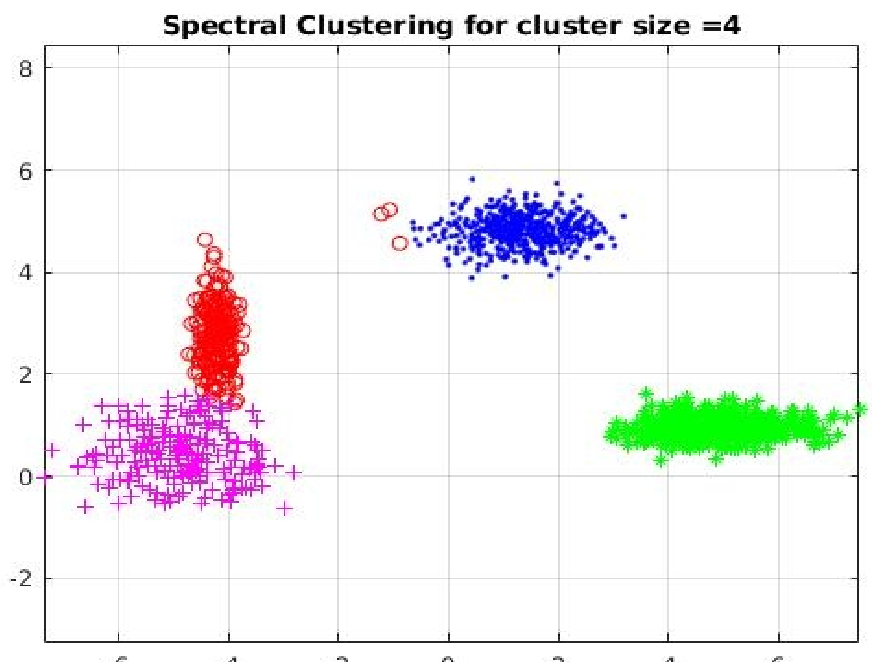
ENEE439M
Assignment 5

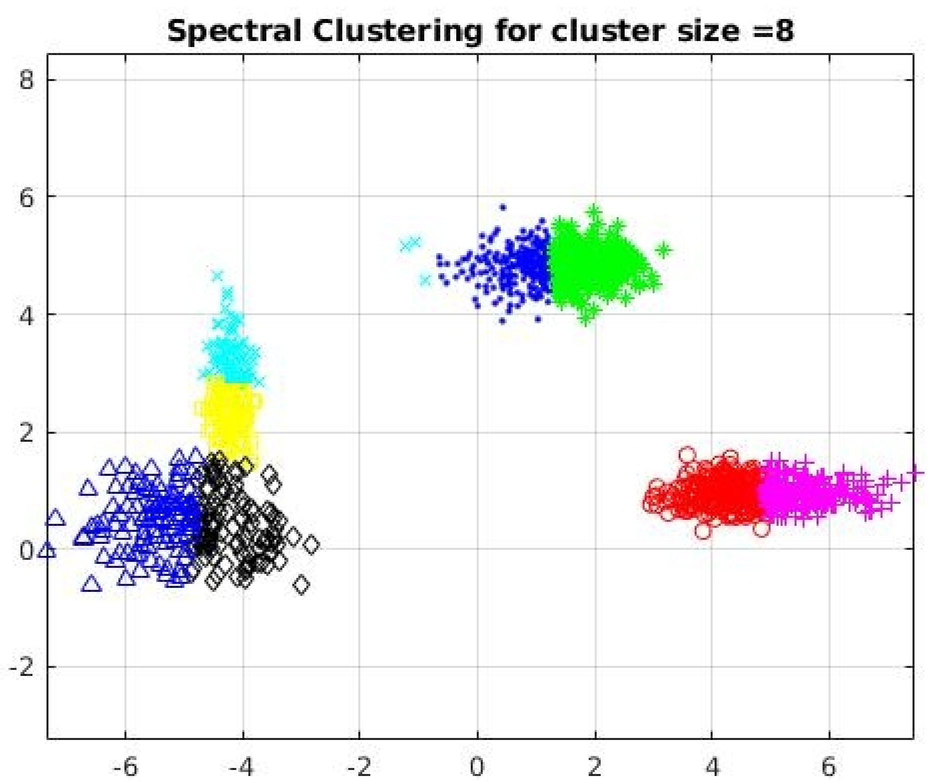


Dataset 2-



ENEE439M
Assignment 5





ENEE439M
Assignment 5

3.

Kmeans									
	Sum Squared Error			Minimum Variance			Determinant Criteria		
Dataset1	195400.2	148589.5	107142.3	131838.3	63351.6	20104.9	3267205650.2	907028711.4	100741842.6
Dataset2	22544.0	2391.3	2534.3	11523.4	955.5	621.3	8107128.4	145456.4	59301.3
	Cluster 2	Cluster 4	Cluster 8	Cluster 2	Cluster 4	Cluster 8	Cluster 2	Cluster 4	Cluster 8

Spectral Clustering									
	Sum Squared Error			Minimum Variance			Determinant Criteria		
Dataset1	207710.9	119774.5	43281.8	207710.9	119774.5	3453045768.3	10751492347.6	2272539154.6	461714215.0
Dataset2	8529.5	982.7	457.4	8529.5	982.7	648331.9	7377369.4	158987.7	46851.2
	Cluster 2	Cluster 4	Cluster 8	Cluster 2	Cluster 4	Cluster 8	Cluster 2	Cluster 4	Cluster 8

Sum Squared Error Criteria -
For dataset 1 and 2 using both K-Means and Spectral clustering, the SSE decreases considerably as the number of clusters increase. That is the performance improves with increased number of cluster. Comparing the Kmeans and Spectral clustering, the spectral clustering performs better as compared to Kmeans clustering. However, for the case of 2 clusters the SSE is higher for Spectral Clustering as the points in the outer rim of the data is spread widely as per SSE formula since it measures the sum of distances of the points from it's means the SSE is higher in that case. However, for dataset 2 the difference can be observed with small SSE values.

Minimum Variance Criteria -
For Dataset 1 we can see from the values that the Kmeans performs better as compared to the SC. The Minimum variance increases as the number of clusters increase for dataset 1 when using SC however, it reduces when using Kmeans.
For dataset 2 SC performs better for 2 clusters but again, Kmeans performs better as the number of clusters is increased. As we know that minimum error criteria uses euclidean distance for the measure of similarity, the trend becomes intutive looking at the data.

Determinant Criteria (Jd)-
For dataset 1 and 2 both the Jd decreases as the number of clusters in increased, that is because the Jd measures the scatter within the class. And looking at data, Kmeans performs better than SC for Dataset1, but for dataset 2 SC performs better than Kmeans.

Visual Inference -
Looking at plots from both the methods I think the performance of the algorithms varies based on the

spread of the data and also based on how well defined the dataset is in the space. For certain scenarios Kmeans gave quite good results, but for scenario like donut spread, Kmeans doesn't give required clustering which is quite intutive from visible eyes.

4. Dataset 2 -

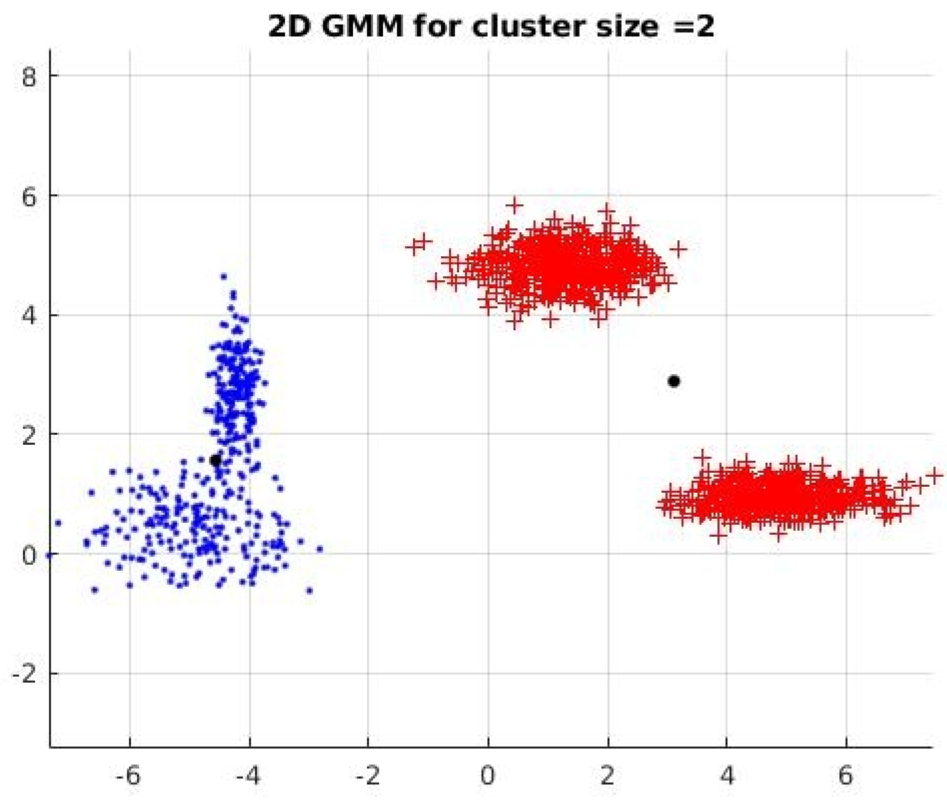
For Cluster Size 2 -

Means -

Mean 1 =	-4.5595	Mean 2 =	3.1122
	1.5676		2.8890

Covariances -

Covariance 1 =	0.5146	0.3590	Covariance 2 =	3.8568	-3.4862
	0.3590	1.6379		-3.4862	3.8061



ENEE439M
Assignment 5

For Cluster Size 4 -

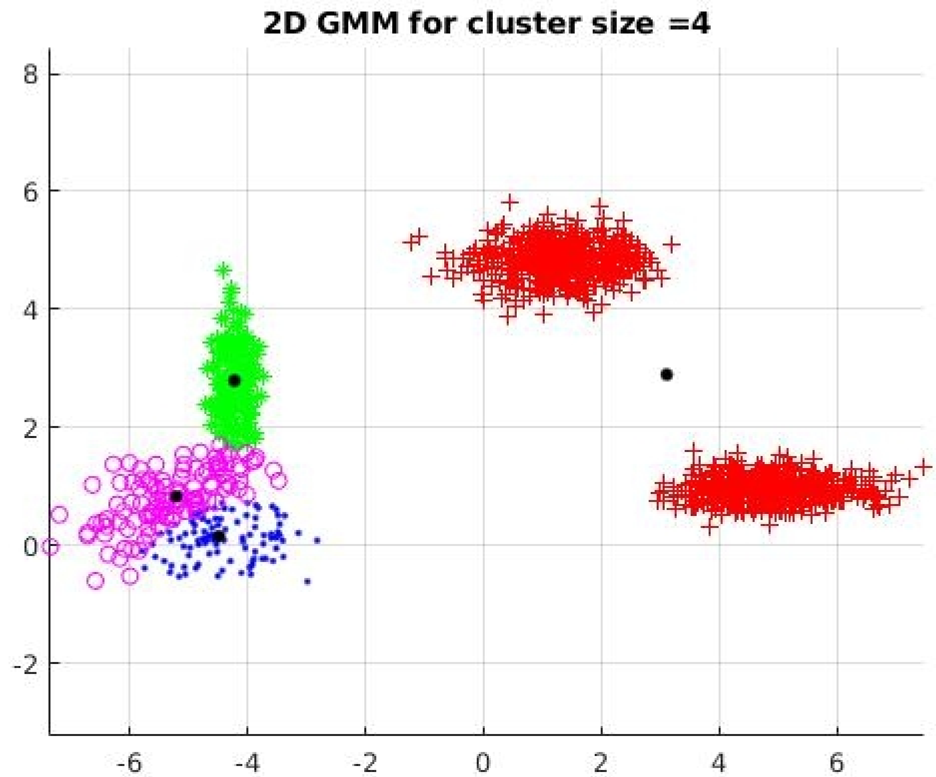
Means -

Mean 1 = -4.4874	Mean 2 = 3.1122	Mean 3 = -4.2235	Mean 4 = -5.2113
0.1539	2.8890	2.7859	0.8277

Covariances -

Covariance 1 = $\begin{matrix} 0.5702 & -0.0080 \\ -0.0080 & 0.1471 \end{matrix}$ Covariance 2 = $\begin{matrix} 3.8568 & -3.4862 \\ -3.4862 & 3.8061 \end{matrix}$ Covariance 3 = $\begin{matrix} 0.0382 & -0.0015 \\ -0.0015 & 0.3325 \end{matrix}$

Covariance 4 = $\begin{matrix} 0.6602 & 0.2464 \\ 0.2464 & 0.2885 \end{matrix}$



ENEE439M
Assignment 5

For Cluster size 8 -

Means -

$$\text{Mean 1} = \begin{matrix} -4.2235 \\ 2.7859 \end{matrix} \quad \text{Mean 2} = \begin{matrix} 4.2445 \\ 1.0307 \end{matrix} \quad \text{Mean 3} = \begin{matrix} 4.4759 \\ 0.8835 \end{matrix} \quad \text{Mean 4} = \begin{matrix} -5.2113 \\ 0.8277 \end{matrix}$$

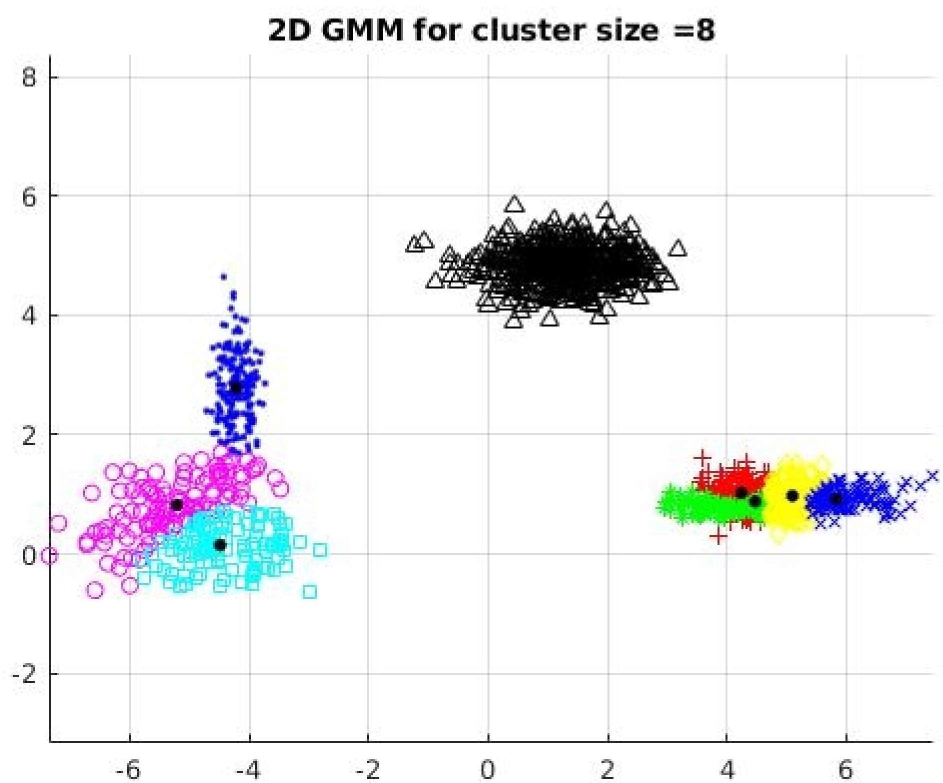
$$\text{Mean 5} = \begin{matrix} -4.4874 \\ 0.1539 \end{matrix} \quad \text{Mean 6} = \begin{matrix} 5.0990 \\ 0.9785 \end{matrix} \quad \text{Mean 7} = \begin{matrix} 1.3084 \\ 4.8230 \end{matrix} \quad \text{Mean 8} = \begin{matrix} 5.8374 \\ 0.9277 \end{matrix}$$

Covariances -

$$\text{Covariance 1} = \begin{matrix} 0.0382 & -0.0015 \\ -0.0015 & 0.3325 \end{matrix} \quad \text{Covariance 2} = \begin{matrix} 0.1310 & -0.0025 \\ -0.0025 & 0.0523 \end{matrix} \quad \text{Covariance 3} = \begin{matrix} 0.5297 & 0.0216 \\ 0.0216 & 0.0232 \end{matrix}$$

$$\text{Covariance 4} = \begin{matrix} 0.6602 & 0.2464 \\ 0.2464 & 0.2885 \end{matrix} \quad \text{Covariance 5} = \begin{matrix} 0.5702 & -0.0080 \\ -0.0080 & 0.1471 \end{matrix} \quad \text{Covariance 6} = \begin{matrix} 0.1422 & 0.0073 \\ 0.0073 & 0.0487 \end{matrix}$$

$$\text{Covariance 7} = \begin{matrix} 0.5265 & 0.0046 \\ 0.0046 & 0.0902 \end{matrix} \quad \text{Covariance 8} = \begin{matrix} 0.3873 & 0.0139 \\ 0.0139 & 0.0290 \end{matrix}$$



Q5. a) Initialisation:—

$$\forall S_K \in \{S_1, S_2\},\ \alpha_1^k = P(x_1 | \pi_1 = S_K)\ P(\pi_1 = S_K)$$

Iteration:—

$$\forall S_K \in \{S_1, S_2\},\ t \in \{2, 3, 4, 5, 6\}.$$

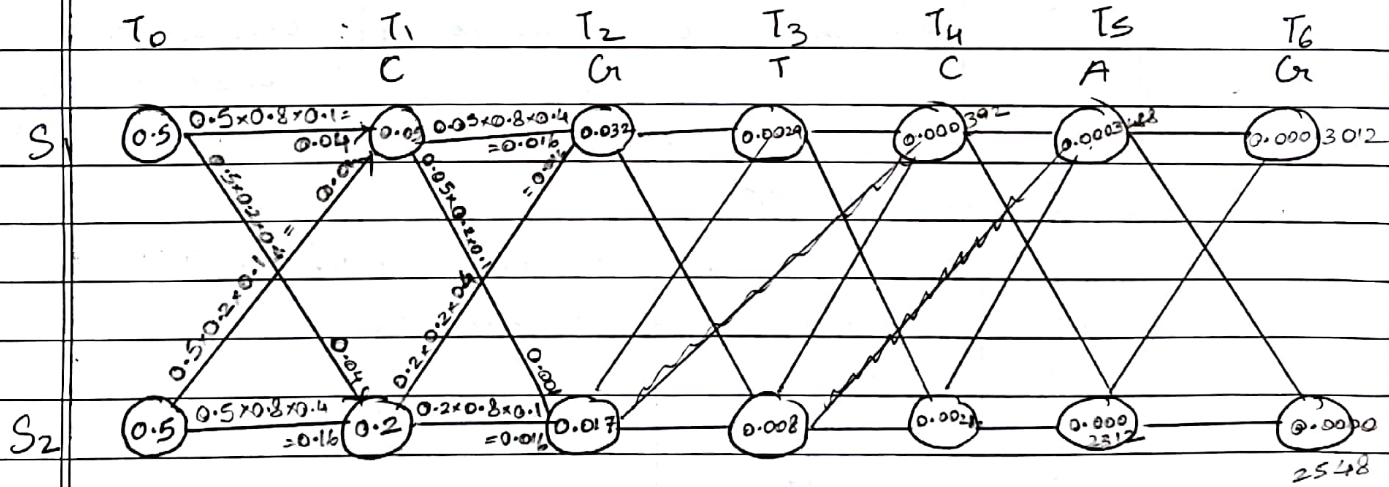
$$\alpha_t^k = P(x_t | \pi_t = S_K) \sum_{i \in \{1,2\}} \alpha_{t-1}\, a_{ik}.$$

Trelis Diagram & Matrices

$$A = \begin{array}{c|cc} & S_1 & S_2 \\ \hline S_1 & 0.8 & 0.2 \\ S_2 & 0.2 & 0.8 \end{array}$$

$$B = \begin{array}{c|cccc} & A & C & G & T \\ \hline S_1 & 0.4 & 0.1 & 0.4 & 0.1 \\ S_2 & 0.1 & 0.4 & 0.1 & 0.4 \end{array}$$

$BC = \ P(S_1) = P(S_2) = 0.5.$



Further Calculations at back.

α	$S_K = S_1$	$S_K = S_2$
α_1^k	$(0.5\times0.8\times0.1)+(0.5\times0.2\times0.1)=0.05$	$(0.5\times0.2\times0.4)+(0.5\times0.8\times0.4)$ $0.04+0.16 = 0.2$
α_2^k	$(0.05\times0.8\times0.4)+(0.2\times0.2\times0.4)=0.032$	$(0.05\times0.2\times0.1)+(0.2\times0.8\times0.1)$ $=0.017$
α_3^k	$(0.032\times0.8\times0.1)+(0.017\times0.2\times0.1)=0.0029$	$(0.032\times0.2\times0.4)+(0.17\times0.8\times0.4)$ $=0.008$
α_4^k	$(0.0029\times0.08\times0.1)+(0.008+0.8\times0.4)$ $=0.000392$	$(0.0029\times0.2\times0.4)+(0.008\times0.8\times0.4)$ $=0.00279$
α_5^k	$(0.000392\times0.8\times0.4)+(0.0028\times0.2\times0.4)$ $=0.0003488$	$(0.000392\times0.2\times0.1)+(0.0028\times0.8\times0.4)$ $=0.0002312$
α_6^k	$(0.0003488\times0.8\times0.4)+(0.0002312\times0.2\times0.4)$ $=0.00013012$	$(0.000348\times0.2\times0.1)+(0.0002312\times0.8\times0.1)=$ 0.00002548

$\therefore$ Final Table

α	$S_K=S_1$	$S_K=S_2$
α_1^k	0.05	0.2
α_2^k	0.032	0.017
α_3^k	0.0029	0.008
α_4^k	0.000392	0.00279
α_5^k	0.000348	0.0002312
α_6^k	0.000130	0.00002548

$$\therefore \quad P(x|M) = 0.00013 + 0.00002548$$
$$= 0.000155$$

Q.5. b) Backward algorithm.

initialise $\omega(T)$, $t = T$, a_{ij}, b_{jk}, visible sequence V^T

for $t \leftarrow t-1$; c

$$\beta_j(t) \leftarrow \sum_{i=1} \beta_i(t+1)\, a_{ij}\, b_{jk}\, v(t+1)$$

until $t = 1$

return $P(V^T) \leftarrow \beta_i(0)$ for the known initial state

end.

$\Rightarrow$ The initialisations:— $\beta_6^1 = 1$, $\beta_6^2 = 1$

And the A & B matrix are same as above.

Calculating $\beta_j(t) \leftarrow \sum_{i=1} \beta_i(t+1)\, a_{ij}\, b_{jk}\, v(t+1)$ for each step.

β	$S_k = S_1$	$S_k = S_2$
β_6^k	1	1
β_5^k	$(1\times0.8\times0.4)+(1\times0.2\times0.1)=0.34$	$(1\times0.2\times0.2)+(1\times0.1\times0.8)=0.16$
β_4^k	$(0.34\times0.4\times0.8)+(0.16\times0.1\times0.2)$ $=0.112$	$(0.34\times0.2\times0.4)+(0.16\times0.8\times0.1)$ $=0.04$
β_3^k	$(0.112\times0.1\times0.2)+(0.04\times0.8\times0.4)$ $=0.01216$	$(0.112\times0.1\times0.2)+(0.04\times0.4\times0.2)$ $=0.01504$
β_2^k	$(0.01216\times0.8\times0.1)+(0.01504\times0.2\times0.4)$ $=0.002176$	$(0.01216\times0.2\times0.1)+(0.01504\times0.8\times0.4)$ $=0.005056$
β_1^k	$(0.002176\times0.8\times0.4)+(0.005056\times0.1\times0.2)$ $=0.0007974$	$(0.002176\times0.2\times0.4)+(0.005056\times0.1\times0.8)$ $=0.000579$

Now, using,

$$P(\pi_i = S_1 | x, M) = \frac{\alpha_i^1 \beta_i^1}{\sum_{k \in \{1,2\}} \alpha_i^k \beta_i^k}$$

$P(\pi_1 = S_1 | x, M) = 0.3987 / (0.3987 + 1.158)$
$= 0.25622$

$P(\pi_2 = S_1 | x, M) = (0.696) / (0.696 + 0.8585)$
$= 0.4477$

$P(\pi_3 = S_1 | x, M) = (0.3526) / (0.3526 + 1.2032)$
$= 0.2266$

$P(\pi_4 = S_1 / x, M) = (0.4383) / (0.383 + 1.1168)$
$= 0.2834.$

$P(\pi_5 = S_1 / x, M) = (1.185) / (1.185 + 0.37)$
$= 0.762$

$P(\pi_6 = S_1 / x, M) = (1.30112) / (1.30112 + 0.2547)$
$= 0.837.$

Q.5. c) Using the viterbi Algorithm.

$\delta_1(1) = \pi_1 b_1(O_1) = 0.05.$
$\delta_1(2) = \pi_2 b_2(O_1) = 0.2$
$\psi_1(1) = \psi_1(2) = S_1$

$\delta_2(1) = \max(0.05 \times 0.8, 0.2 \times 0.2) \times 0.4$
$= 0.016.$ $\psi_2(1) = S_1$
$\delta_2(2) = \max(0.05 \times 0.2, 0.2 \times 0.8) \times 0.1$ $\psi_2(2) = S_2$
$= 0.016.$

$\delta_3(1) = \max(0.016 \times 0.8, 0.016 \times 0.2) \times 0.1$
$= 0.00128$ $\psi_3(1) = S_1$
$\delta_3(2) = \max(0.016 \times 0.2, 0.016 \times 0.8) \times 0.4$
$= 0.00512$
$\psi_3(2) = S_2$

$\delta_4(1) = \max(\boxed{0.00128 \times 0.8}, 0.00512 \times 0.2) \times 0.1$
$= 0.0001024$ $\psi_4(1) = S_1$
$\delta_4(2) = \max(0.00128 \times 0.2, 0.00512 \times 0.8) \times 0.4$
$= 0.001638.$
$\psi_4(2) = S_2$

$\delta_5(1) = \max(0.0001024 \times 0.8, 0.001638 \times 0.2) \times 0.4.$
$= 0.0001311$ $\psi_5(1) = S_2.$
$\delta_5(2) = \max(0.0001024 \times 0.2, 0.01638 \times 0.8) \times 0.1$
$= 0.0001311$ $\psi_5(1) = S_2.$

$\delta_6(1) = \max(0.0001311 \times 0.8, 0.0001311 \times 0.2) \times 0.4$
$= 0.00004195 \quad \psi_6(1) = S_1$
$\delta_6(2) = \max(0.0001311 \times 0.2, 0.0001311 \times 0.8) \times 0.1$
$= 0.0000105 \quad \psi_6(2) = S_2.$

$\therefore$ The optimal states are :-
$S_2 \; S_2 \; S_2 \; S_2 \; S_1 \; S_1$