Problem 1

1(a)

For a particular position t, we have Loss Function as:

$$L(y_t, \hat{y}_t) = \frac{1}{2}||y_t - \hat{y}_t||_2^2 = \frac{1}{2}(y_t - \hat{y}_t)^T(y_t - \hat{y}_t) \tag{1}$$

$$y_t = W_{HO}s_t \tag{2}$$

$$s_t = \sigma(W_{IH}x_t + W_{HH}s_{t-1}) \tag{3}$$

From (1) we find Gradient of L with respect to y_t:

$$\nabla_{y_t}L = \frac{\partial L}{\partial y_t} = y_t - \hat{y}_t \tag{4}$$

1(b)

From Eqns (1) and (2), we get:

$$\nabla_{s_t}L = \frac{\partial L}{\partial y_t} \times \frac{\partial y_t}{\partial s_t}$$

$$= (y_t - \hat{y}_t)W_{HO}^T$$

$$\Rightarrow \nabla_{s_t}L = W_{HO}^T(W_{HO}s_t - \hat{y}_t) \tag{5}$$

Lets now express $\nabla_{s_t}L$ in terms of $\nabla_{s_{t+1}}L$ using eqn (3) :

$$\nabla_{s_t}L = \frac{\partial L}{\partial s_{t+1}} \times \frac{\partial s_{t+1}}{\partial s_t}$$

$$= \nabla_{s_{t+1}}L \times \frac{\partial[\sigma(W_{IH}x_{t+1} + W_{HH}s_t)]}{\partial s_t}$$

$$= [W_{HH}^T \nabla_{s_{t+1}}L]\sigma(W_{IH}x_{t+1} + W_{HH}s_t)(1 - \sigma(W_{IH}x_{t+1} + W_{HH}s_t))$$

$$\Rightarrow \nabla_{s_t}L = [W_{HH}^T \nabla_{s_{t+1}}L]\sigma'(W_{IH}x_{t+1} + W_{HH}s_t) \tag{6}$$

1(c)

$$\nabla_{W_{IH}} L = \frac{\partial L}{\partial s_t} \times \frac{\partial s_t}{\partial W_{IH}} = W_{HO}^T (W_{HO} s_t - \hat{y}_t) \times \sigma'(W_{IH} x_{t+1} + W_{HH} s_t) x_t$$

$$= \nabla_{s_t} L \sigma'(W_{IH} x_{t+1} + W_{HH} s_t) x_t$$

$$\nabla_{W_{HH}} L = \frac{\partial L}{\partial s_t} \times \frac{\partial s_t}{\partial W_{HH}} = W_{HO}^T (W_{HO} s_t - \hat{y}_t) \times \sigma'(W_{IH} x_{t+1} + W_{HH} s_t) s_{t-1}$$

$$= \nabla_{s_t} L \sigma'(W_{IH} x_{t+1} + W_{HH} s_t) s_{t-1}$$

$$\nabla_{W_{HO}} L = \frac{\partial L}{\partial y_t} \times \frac{\partial y_t}{\partial W_{HO}} = (y_t - \hat{y}_t) s_t = \nabla_{s_t} L \times s_t$$

1(d)

$$\nabla_{W_{IH}} L = \frac{\partial L}{\partial s_t} \times \frac{\partial s_t}{\partial W_{IH}} = \tau \times W_{HO}^T (W_{HO} s_t - \hat{y}_t) \times \sigma'(W_{IH} x_{t+1} + W_{HH} s_t) x_t$$

$$= \tau \times \nabla_{s_t} L \sigma'(W_{IH} x_{t+1} + W_{HH} s_t) x_t$$

$$\nabla_{W_{HH}} L = \frac{\partial L}{\partial s_t} \times \frac{\partial s_t}{\partial W_{HH}} = \tau \times W_{HO}^T (W_{HO} s_t - \hat{y}_t) \times \sigma'(W_{IH} x_{t+1} + W_{HH} s_t) s_{t-1}$$

$$= \tau \times \nabla_{s_t} L \sigma'(W_{IH} x_{t+1} + W_{HH} s_t) s_{t-1}$$

$$\nabla_{W_{HO}} L = \frac{\partial L}{\partial y_t} \times \frac{\partial y_t}{\partial W_{HO}} = (y_t - \hat{y}_t) s_t = \nabla_{s_t} L \times s_t$$

Problem 2

Given,

$$\tilde{D} = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} ||\phi(x_n) - \tilde{\mu_k}||_2^2 \tag{7}$$

2(a)

Eqn (7) can be represented as :

$$\tilde{D} = \sum_{n=1}^{N}\sum_{k=1}^{K} r_{nk}||\phi(x_n) - \tilde{\mu_k}||_2^2$$

$$= \sum_{n=1}^{N}\sum_{k=1}^{K} r_{nk}[\phi(x_n) - \tilde{\mu_k}]^T[\phi(x_n) - \tilde{\mu_k}]$$

$$= \sum_{n=1}^{N}\sum_{k=1}^{K} r_{nk}\left[\phi(x_n) - \frac{\sum_{i=1}^{N}\phi(x_i)}{\sum_{n=1}^{N} r_{nk}}\right]^T\left[\phi(x_n) - \frac{\sum_{i=1}^{N}\phi(x_i)}{\sum_{n=1}^{N} r_{nk}}\right]$$

$$\left\{\because \tilde{\mu_k} = \frac{\sum_{i=1}^{N}\phi(x_i)}{\sum_{n=1}^{N} r_{nk}}\right\}$$

$$= \sum_{n=1}^{N}\sum_{k=1}^{K} r_{nk}\left\{\phi(x_n)^T.\phi(x_n) - \frac{2\sum_{i=1}^{N}\phi(x_n)^T.\phi(x_i)}{\sum_{n=1}^{N} r_{nk}} + \sum_{i=1}^{N}\frac{\phi(x_i)^T.\phi(x_i)}{(\sum_{n=1}^{N} r_{nk})^2}\right\}$$

$$\tilde{D} = \sum_{n=1}^{N}\sum_{k=1}^{K} r_{nk}\left\{k(x_n, x_n) - \frac{2\sum_{i=1}^{N}k(x_n, x_i)}{\sum_{n=1}^{N} r_{nk}} + \sum_{i=1}^{N}\frac{k(x_i, x_i))}{(\sum_{n=1}^{N} r_{nk})^2}\right\} \tag{8}$$

Hence we show that $\tilde{D}$ can be represented in terms of only Kernel $k(x_i, x_j) = \phi(x_i)^T.\phi(x_j)$

2(b)

When we get a data point, we find the distance of the data point to each cluster-mean and assign it to the cluster that it is closest to.
μ_k: Mean of Cluster k
x: Data point
C_j: Cluster j
$\phi(x)$: Data point x transformed in space
Equation of assigning a data point to a cluster k is given by μ_k where :

$$k = \text{arg min}_j||x - \mu_j||_2^2$$

$$= \text{arg min}_j||\phi(x) - \frac{\sum_{x_n \in C_j}\phi(x_n)}{|x_n \in C_j|}||_2^2$$

$$= \text{arg min}_j\left[\phi(x)^T.\phi(x) - \frac{2\sum_{x_n \in C_j}\phi(x_n)^T.\phi(x)}{|x_n \in C_j|} + \frac{\sum_{x_n \in C_j}\phi(x_n)^T.\phi(x_n)}{|x_n \in C_j|^2}\right]$$

$$k = \text{arg min}_j\left[k(x, x) - \frac{2\sum_{x_n \in C_j}k(x_n, x)}{|x_n \in C_j|} + \frac{\sum_{x_n \in C_j}k(x_n, x_n)}{|x_n \in C_j|^2}\right]$$

$$\mu_k = \text{arg min}_j \left[k(x,x) - \frac{2\sum_{x_n \in C_j} k(x_n, x)}{|x_n \in C_j|} + \frac{\sum_{x_n \in C_j} k(x_n, x_n)}{|x_n \in C_j|^2} \right] \tag{9}$$

2(c)

Kernel K-means Algorithm
Input:

K: Kernel Matrix
$C_1, \dots c_k$: k clusters
k: Number of clusters

Output:

Final grouped clusters $C_1, \dots C_k$

Pseudo-code

1. Randomly partition points $x_{1,\dots,n}$ into k clusters $C_1, C_2, \dots, c_k$
2. **For all** points x_n $\quad n = 1, 2, \dots, N$ **do**
3. **For all** clusters C_k $\quad k = 1, 2, \dots, K$ **do**
4. Compute $||\phi(x_n) - \mu_k||^2$ using eqn (8)
5. **end for**
6. Find $C^*(x_n) = arg\ min_k ||\phi(x_n) - \mu_k||^2$
7. **end for**
8. **For all** clusters C_k $\quad k = 1, 2, \dots, k$ **do**
9. Update clusters $C_k = \{x_n | C^*(x_n) = k\}$
10. **end for**
11. **If** Converged **then**
12. **return** Clusters
13. **else**
14. **goto** Step 2
15. **end if**

Problem 3

Given Zero-inflated Poisson Distribution:

$$p(x_i) = \begin{cases} \pi + (1-\pi)e^{-\lambda} & \text{If } x_i = 0 \\ (1-\pi)\frac{\lambda^{x_i} e^{-\lambda}}{x_i!} & \text{If } x_i > 0 \end{cases} \tag{10}$$

The Incomplete Log-Likelihood is given by :

$$l(\pi, \lambda; x) = \sum_{x_i=0} \ln\left(\pi + (1-\pi)e^{-\lambda}\right) + \sum_{x_i>0} \ln(1-\pi) + \sum_{x_i>0} x_i \ln\lambda - \sum_{x_i>0} \lambda - \sum_{x_i>0} \ln(x_i!) \tag{11}$$

Since we cant reduce the above equation to simple function, we cant maximize the likelihood. So, we use EM Algorithm to solve such problems.

3(a)

Let $Z = z_1, z_2, \ldots, z_n$ be the Hidden variables. Then we define :

$$z_i = \begin{cases} 1 & \text{If } x_i \text{ is from Zero state} \\ 0 & \text{If } x_i \text{ is from Poisson state} \end{cases} \tag{12}$$

Using value z, the new log likelihood is :

$$L1(\pi, \lambda; x, z) = \prod_{x_i=0} \left[z_i\pi + (1-z_i)(1-\pi)e^{-\lambda}\right]$$
$$L2(\pi, \lambda; x, z) = \prod_{x_i>0} \left[(1-z_i)(1-\pi)\frac{\lambda^{x_i}e^{-\lambda}}{x_i!}\right]$$

Applying log on both sides and adding L1 and L2:

$$l(\pi, \lambda; x, z) = \sum_{i=0}^{n} \ln(z_i\pi) + \sum_{i=0}^{n} \ln(1-z_i) + \sum_{i=0}^{n} \ln(1-\pi) + \sum_{i=0}^{n}(x_i\lambda - \lambda - \ln(x_i!)) \tag{13}$$

Eqn(13) gives the Complete log-likelihood function where each parameter can be easily separated.

3(b)

E-Step:

$$p(z_i = 1|x_i; \pi, \lambda) = \frac{P(x_i|\text{zero state})P(\text{zero state})}{P(x_i|\text{zero state})P(\text{zero state}) + P(x_i|\text{poisson state})P(\text{poisson state})} \tag{14}$$

$$p(z_i|x_i = 0; \pi, \lambda) = \frac{\pi}{\pi + (1-\pi)e^{-\lambda}}$$
$$p(z_i|x_i > 0; \pi, \lambda) = 0 \qquad \{\because P(x_i > 0|\text{zero state}) = 0\}$$
$$p(z_i|x_i; \pi, \lambda) = \begin{cases} \frac{\pi}{\pi+(1-\pi)e^{-\lambda}} & \text{If } x_i = 0 \\ 0 & \text{If } x_i > 0 \end{cases} \tag{15}$$

M-Step:

$$\gamma_{ik} = p(z_i|x_i; \pi, \lambda)$$
$$Q(\theta, \theta^{old}) = \sum_i \sum_k \gamma_{ik} log\left(\left(\frac{\pi}{\pi + (1-\pi)e^{-\lambda}}\right)^{x_k} \left(\frac{(1-\pi)e^{-\lambda}}{\pi + (1-\pi)e^{-\lambda}}\right)^{1-x_k}\right)$$
$$\frac{dQ(\theta, \theta^{old})}{d\pi} = 0, \qquad \Rightarrow$$

Problem 4

4.1

600 Blob points and 500 Circle points were loaded to MATLAB environment.

4.2

(a)

Graphs are plotted in the figures below.

(b)

We see that K-Means work fine for Blob but it fails for Circles data set. It is because, for Circles data set, we cant linearly separate the data on a 2-d plane. So, we need to project the data to a higher degree and use a hyperplane to separate the clusters.

4.3

(a)

I used RBF kernel

(b)

4.4

(a)

Graphs for all 5 runs are plotted below.

(b)

The Mean and Co-variance for all 3 GMMs are:

μ:

	X	**Y**
GMM1	0.7590	0.6798
GMM2	-0.3259	0.9713
GMM3	-0.6395	1.4746

Co-variance:
GMM1:

$$\begin{matrix} 0.0272 & -0.0084 \\ -0.0084 & 0.0404 \end{matrix}$$

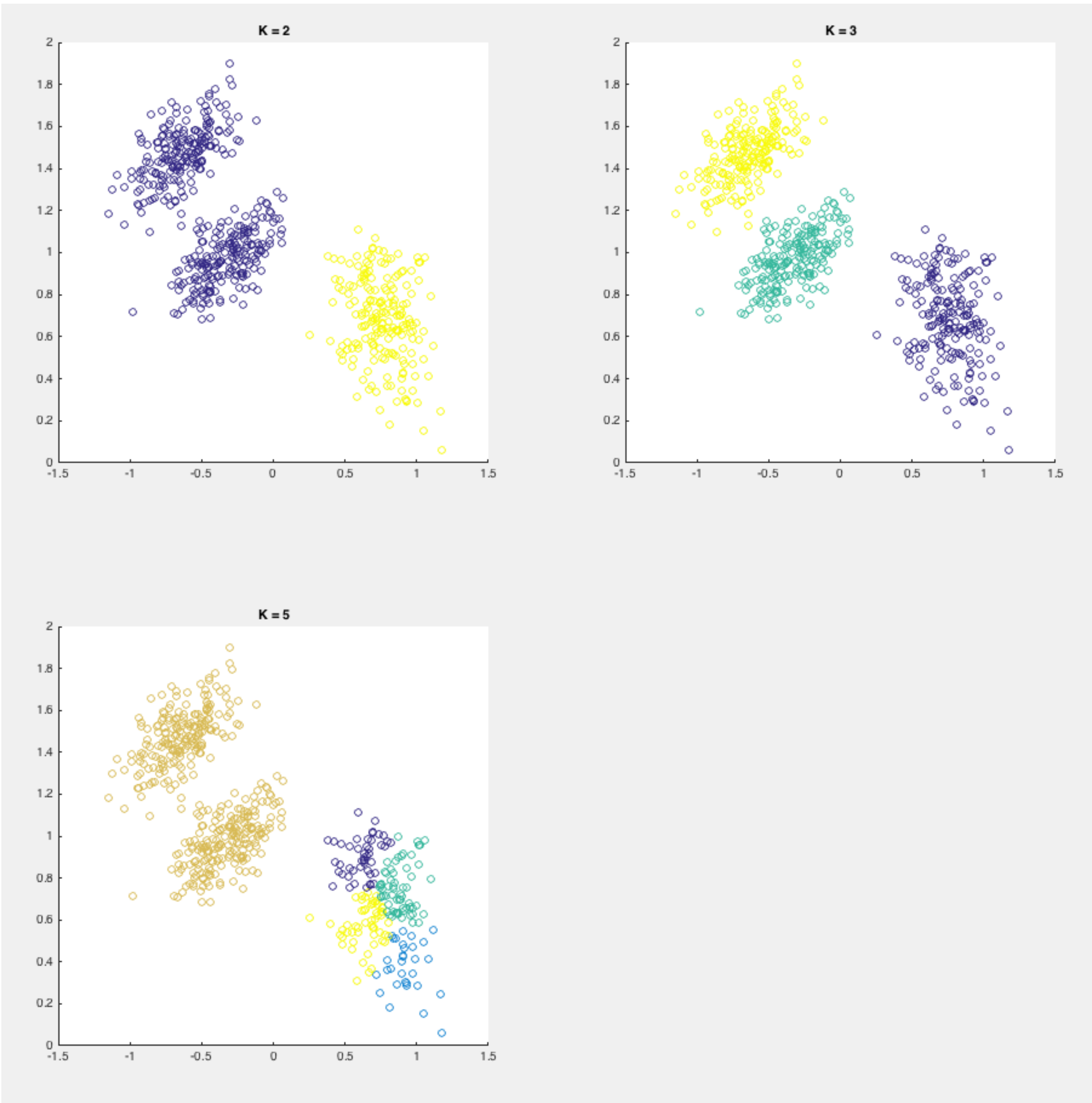

Figure 1: 4.2(a) Clusters for Blob data

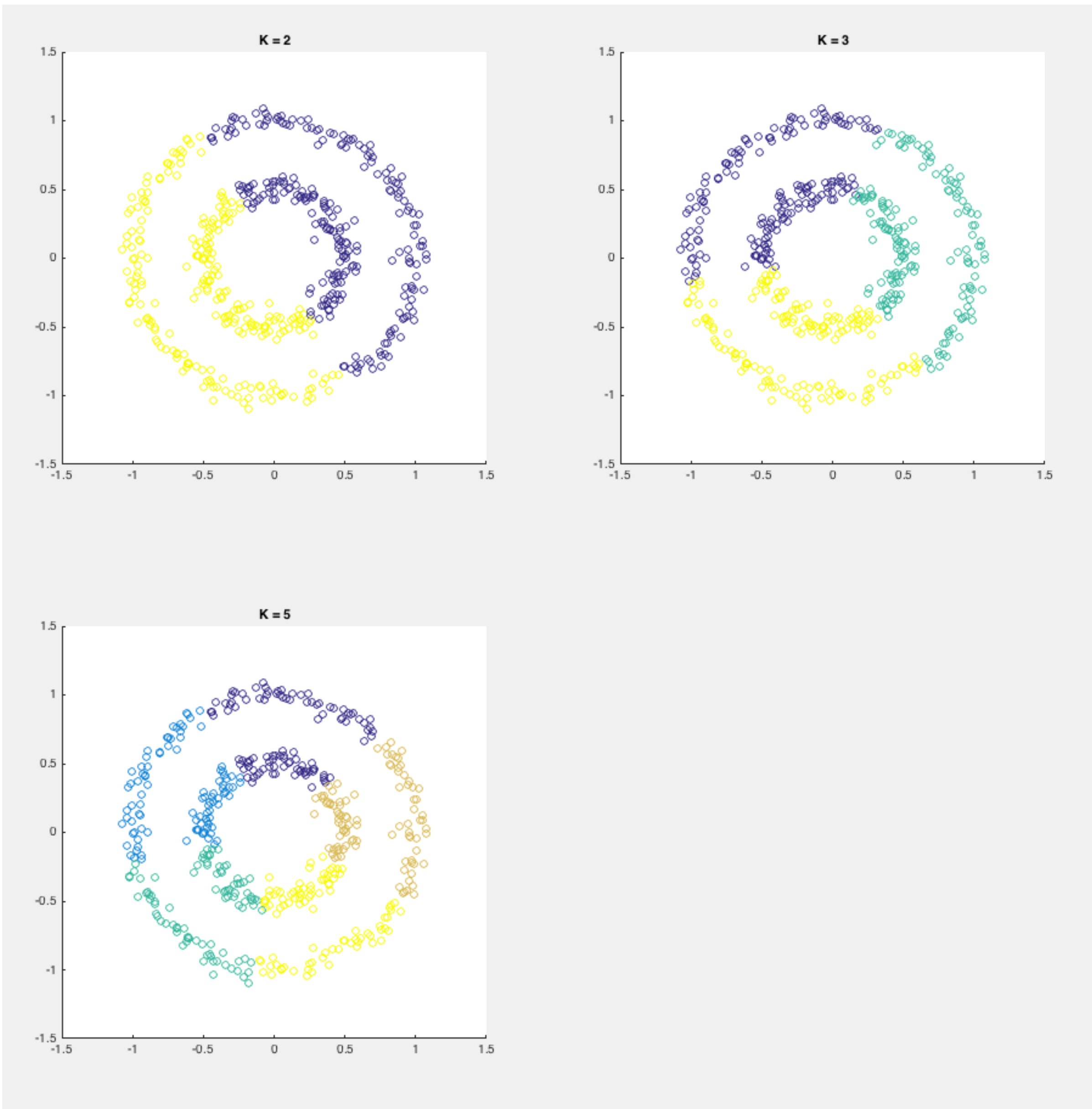

Figure 2: 4.2(a) Clusters for Circle data

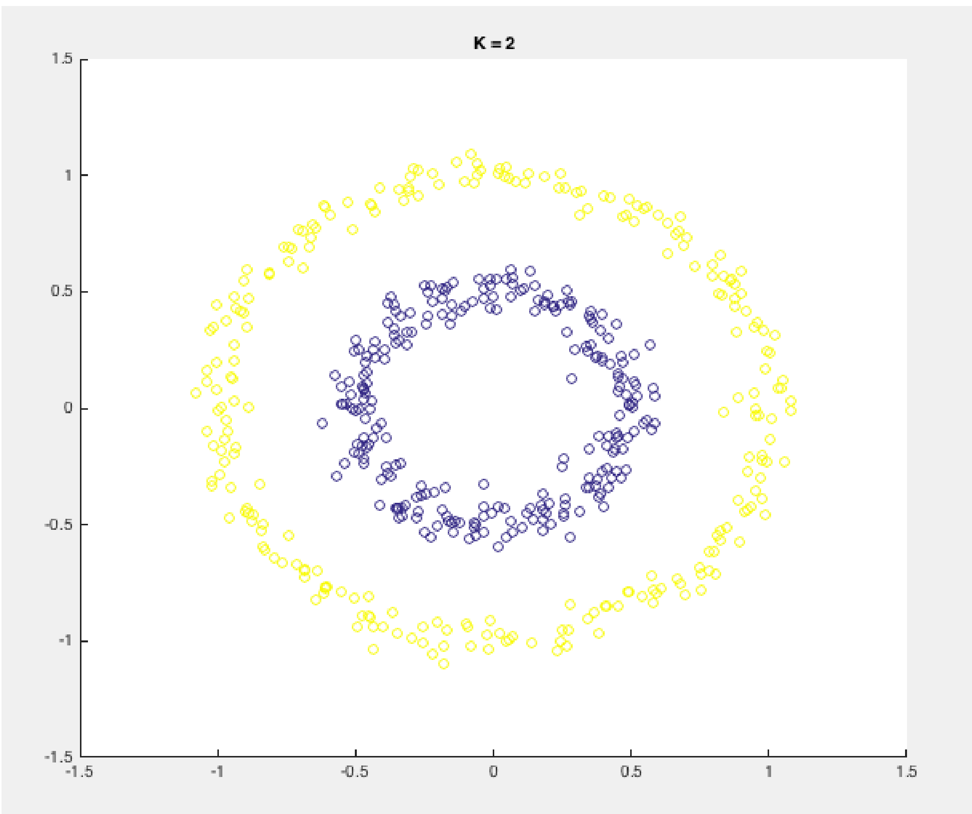

Figure 3: 4.3(b) RBF Kernel clusters for Circle data

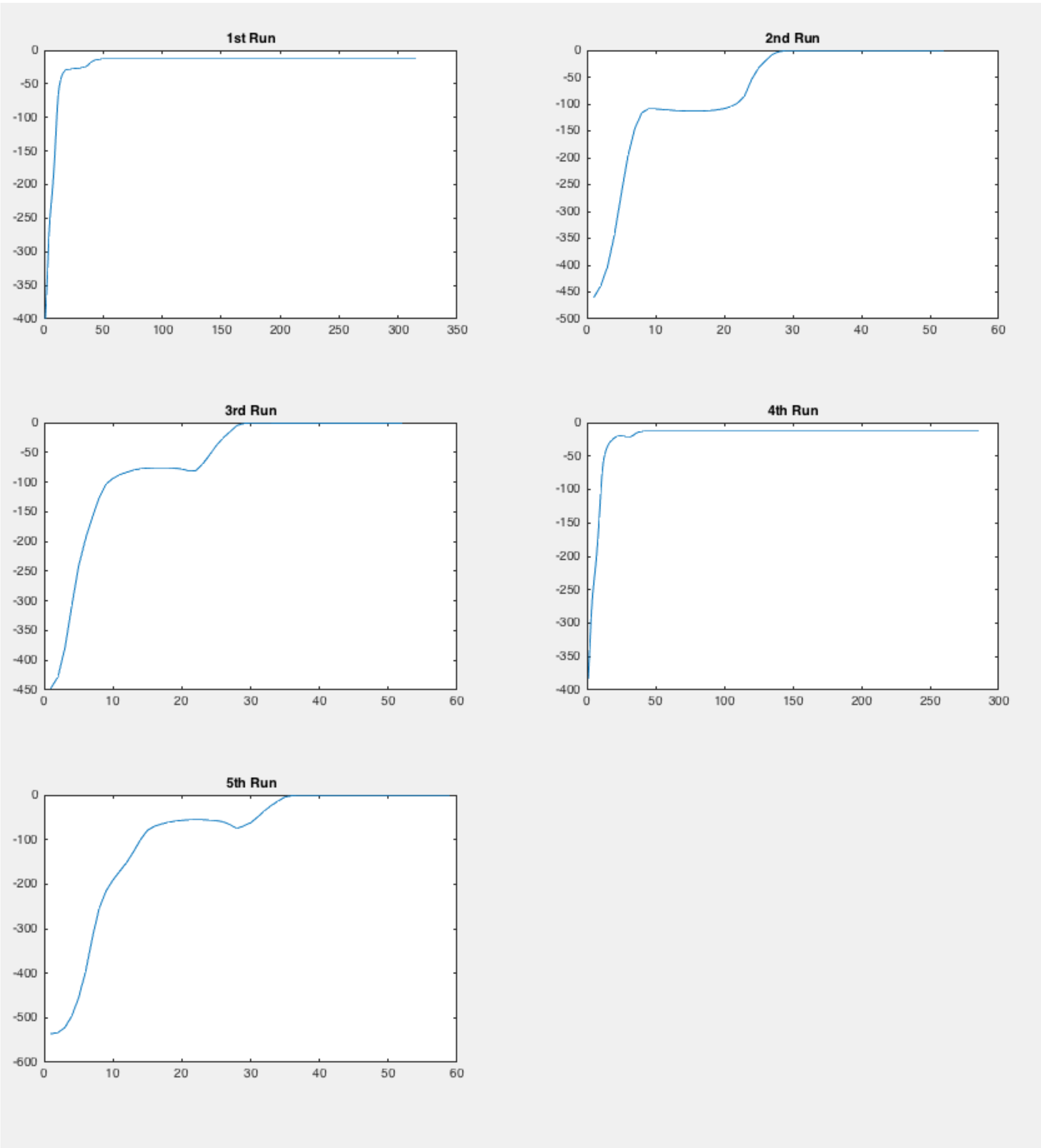

Figure 4: 4.4(a) GMM convergence plots against log likelihood

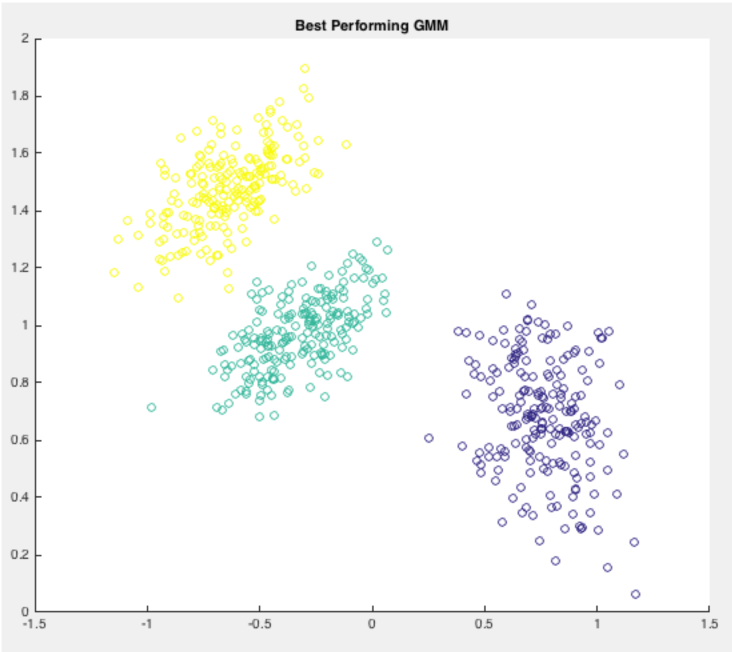

Figure 5: 4.4(b) Clustering Blob data using best performing GMM

GMM2:

0.0360	0.0146
0.0146	0.0163

GMM3:

0.0360	0.0155
0.0155	0.0194