Report: PRML Assignment 1

Keval Dodiya (CS19M023)

Karan Jivani (CS19M030)

Problem 1

Part 1: Plot the Dataset.

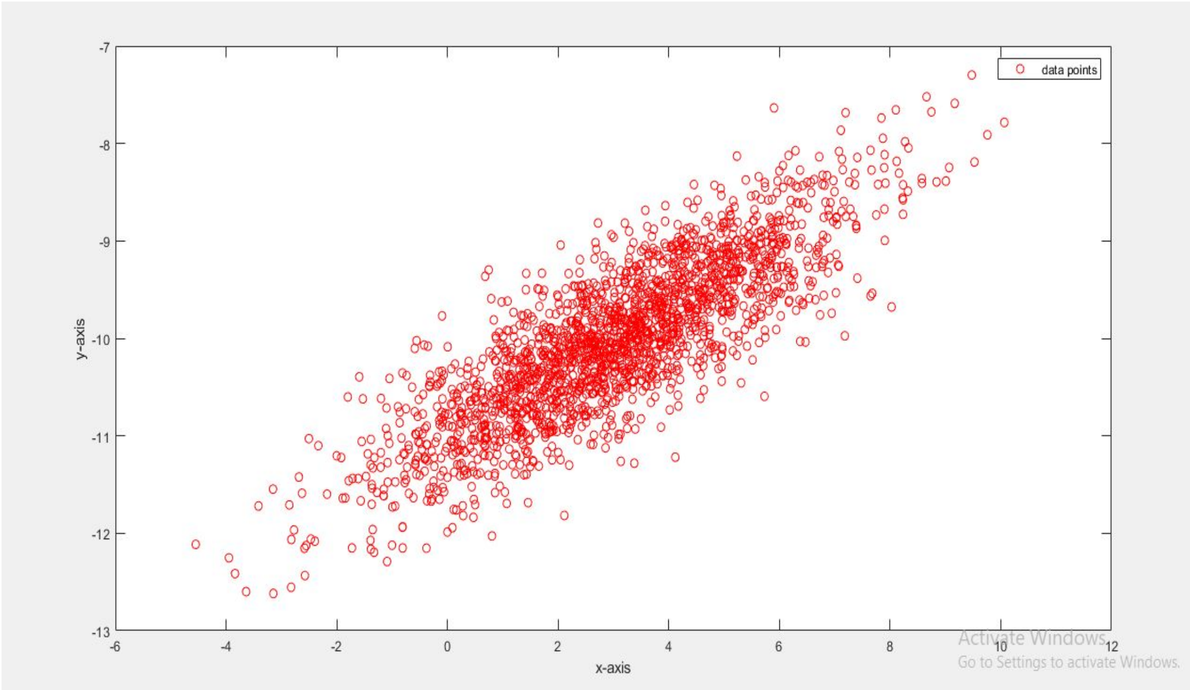


Figure 1.1: Scatter Plot of Dataset1.csv

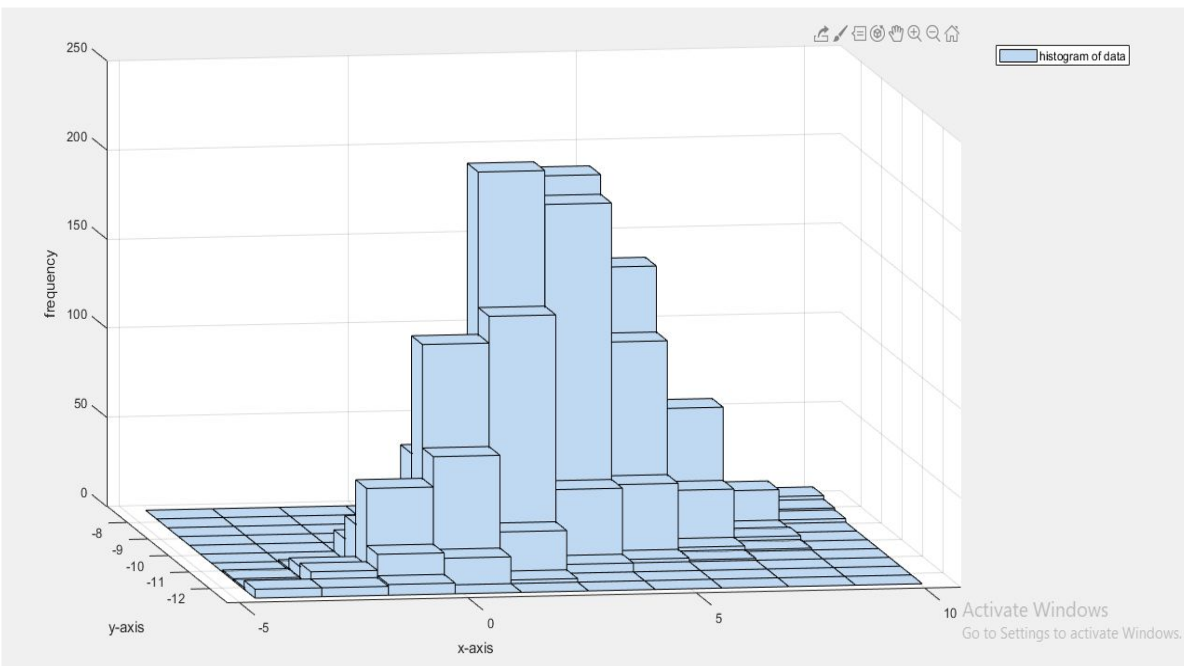


Figure 1.2: Histogram of Frequency vs *(x, y)*

From Fig. (1.1) and Fig. (1.2) Given Data might be generated from a Bivariate Normal Distribution [Eq. (1.1)], since in Histogram, Frequency is Higher near Center of the graph while decreasing towards other directions. To make this more clear, Fig. (1.3) shows Histograms of X-coordinate and Y-coordinate independently.

$$p(x; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \quad \text{--- (1.1)}$$

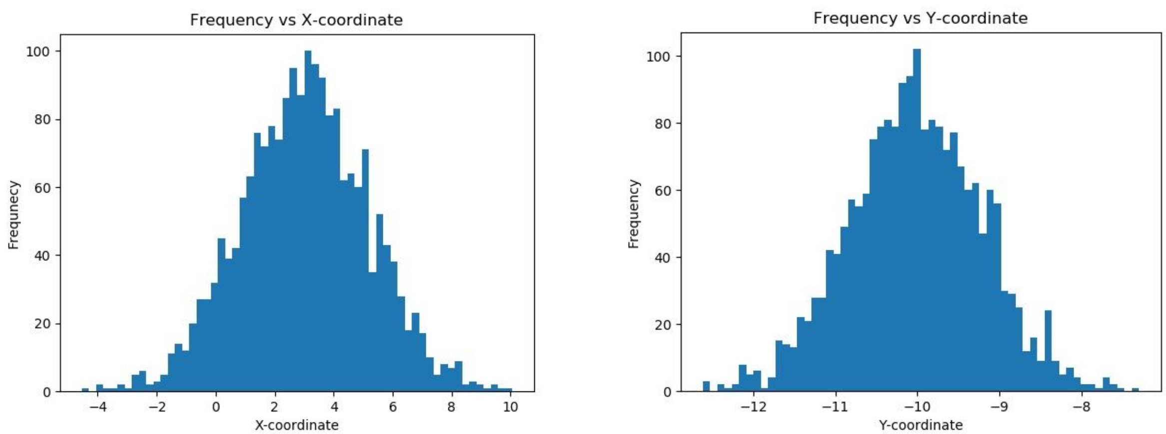


Figure 1.3: Independent Histograms of X and Y Coordinates

Part 2: Find the maximum likelihood estimator of the parameters of the distribution.

Coordinate	Mean (μ)	Standard Deviation (σ)
X	3.013867	2.153646
Y	-10.018809	0.809753

Note that, Covariance of the data is: 1.5703

Part 3: Find the log-likelihood value.

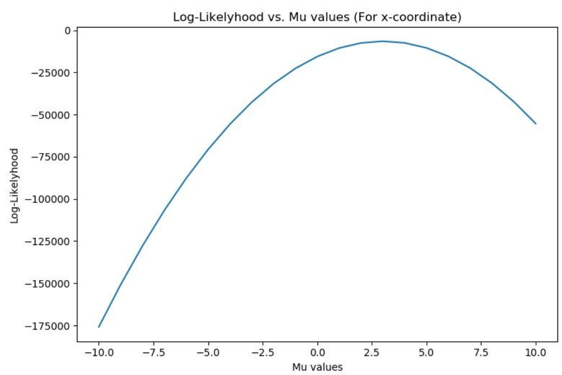
- Log Likelihood Value for Maximum Likelihood Estimator : -5598.9
- Log Likelihood for maximum estimator for ‘x’ (Calculating Independently) with Mean : 3.013867 and Standard Deviation : 2.153646 is : -4372.201594
- Log Likelihood for maximum estimator for ‘y’ (Calculating Independently) with Mean : -10.018809 and Standard Deviation : 0.809753 is : -2415.825657

Part 4: Find the maximum likelihood estimator for mean if variance of data is 1 and covariance is 0.

- Since the calculations of MLE of mean doesn't depend on the value of the covariance, MLE for mean doesn't change.

Coordinate	Mean (μ)
X	3.013867
Y	-10.018809

Part 5: Plot the log-likelihood of the dataset as a function of mean under the assumptions (Variance = 1, Covariance = 0) and vary the values of each component of mean in {-10, … , 10}.



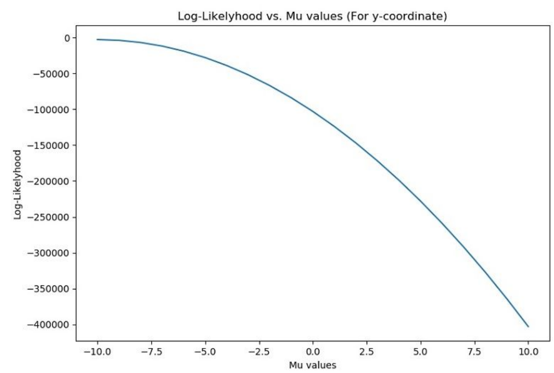
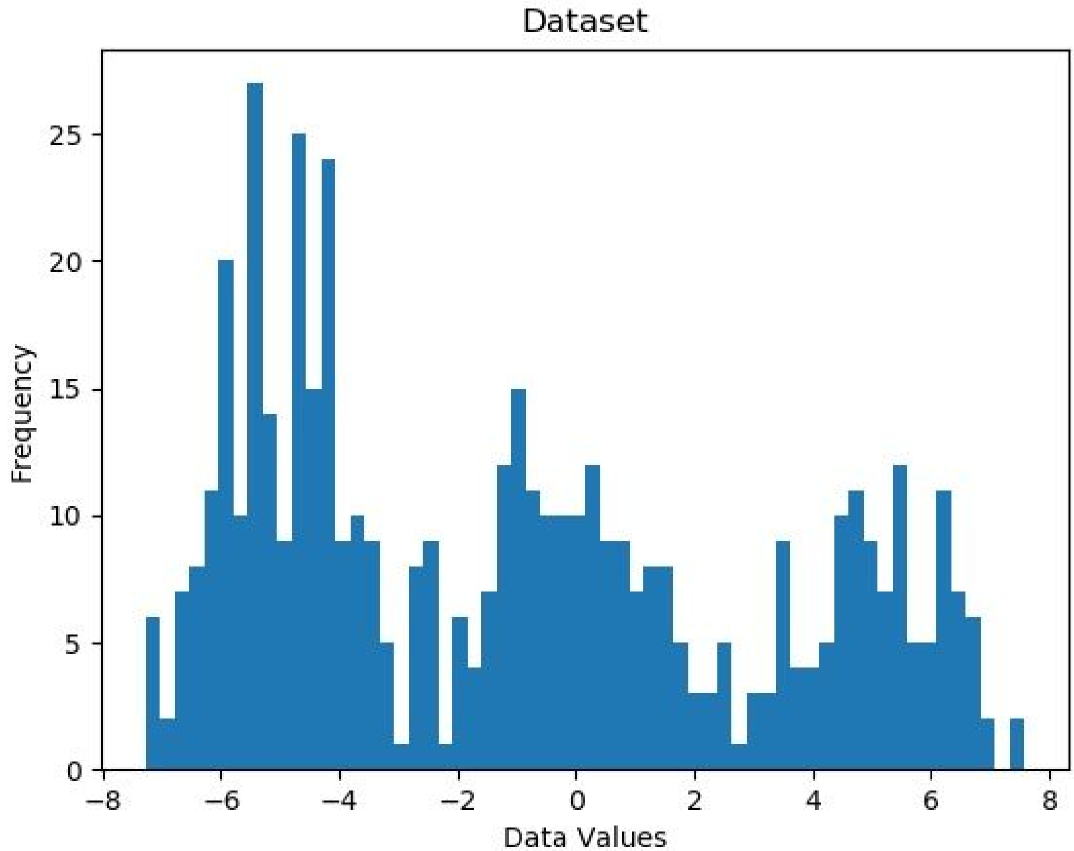


Figure 1.4: Log-Likelihood vs Mean Values for X and Y Coordinates

- From Fig. (1.4), it can be seen that, maximum value of log-likelihood occurs for when mean equals to maximum likelihood estimate value (i.e. 3.013867 for x-coordinate and -10.018809 for y-coordinate).

Problem 2



Part 2: Run GMM for k clusters where $k \in \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ and tabulate the parameters.

For $k = 1$,

Cluster Number	Mean (μ)	Standard Deviation (σ)	Fraction per Cluster (π)
1	-1.032890	4.124402	1.0

For $k = 2$,

Cluster Number	Mean (μ)	Standard Deviation (σ)	Fraction per Cluster (π)
1	1.753850	2.987656	0.594
2	-5.110042	0.923997	0.406

For $k = 3$,

Cluster Number	Mean (μ)	Standard Deviation (σ)	Fraction per Cluster (π)
1	-0.271885	1.977127	0.434
2	-5.237852	0.847772	0.374
3	5.437838	0.859906	0.192

For $k = 4$,

Cluster Number	Mean (μ)	Standard Deviation (σ)	Fraction per Cluster (π)
1	-1.694751	1.209294	0.256
2	4.916695	1.249725	0.248
3	-5.260251	0.836154	0.368
4	0.917172	0.543713	0.128

For $k = 5$,

Cluster Number	Mean (μ)	Standard Deviation (σ)	Fraction per Cluster (π)
1	3.994396	0.936263	0.152
2	-4.836837	1.152149	0.46
3	0.649554	0.608621	0.156
4	-1.079560	0.436232	0.13
5	6.116856	0.547037	0.102

For $k = 6$,

Cluster Number	Mean (μ)	Standard Deviation (σ)	Fraction per Cluster (π)
1	4.128802	1.855843	0.318
2	-5.825453	0.575048	0.22
3	-4.166050	0.122133	0.072
4	-0.544612	0.744947	0.222
5	-4.641153	0.101607	0.078
6	-3.126444	0.497816	0.09

For $k = 7$,

Cluster Number	Mean (μ)	Standard Deviation (σ)	Fraction per Cluster (π)
1	0.685046	0.795786	0.144
2	-4.836837	1.152149	0.46
3	-1.483958	0.290845	0.058

4	-0.753796	0.193884	0.072
5	1.630105	0.192307	0.036
6	4.305845	0.644185	0.128
7	6.116856	0.547037	0.102

For $k = 8$,

Cluster Number	Mean (μ)	Standard Deviation (σ)	Fraction per Cluster (π)
1	-3.915105	0.846197	0.27
2	-5.898031	0.543571	0.202
3	0.211293	0.307320	0.092
4	1.279555	0.306800	0.064
5	2.602811	0.436649	0.036
6	5.486423	0.829293	0.186
7	3.652862	0.173893	0.032
8	-0.989894	0.349645	0.118

For $k = 9$,

Cluster Number	Mean (μ)	Standard Deviation (σ)	Fraction per Cluster (π)
1	-5.110042	0.923997	0.406
2	-2.500534	0.530529	0.076
3	0.536801	0.271223	0.078
4	5.986484	0.606175	0.118
5	-1.066826	0.228186	0.08

6	-0.315561	0.230179	0.062
7	4.635600	0.220339	0.066
8	1.830888	0.544129	0.072
9	3.587842	0.254820	0.042

For $k = 10$,

Cluster Number	Mean (μ)	Standard Deviation (σ)	Fraction per Cluster (π)
1	4.635600	0.220339	0.066
2	2.459611	0.136889	0.018
3	0.697885	0.646112	0.162
4	-1.961283	0.054173	0.012
5	-4.160284	0.104046	0.066
6	-0.989894	0.349645	0.118
7	-4.776803	0.241712	0.12
8	5.299500	1.224420	0.164
9	-5.977896	0.513172	0.182
10	-3.143184	0.505019	0.092

Part 3: Plot the log-likelihood vs. number of clusters for dataset.

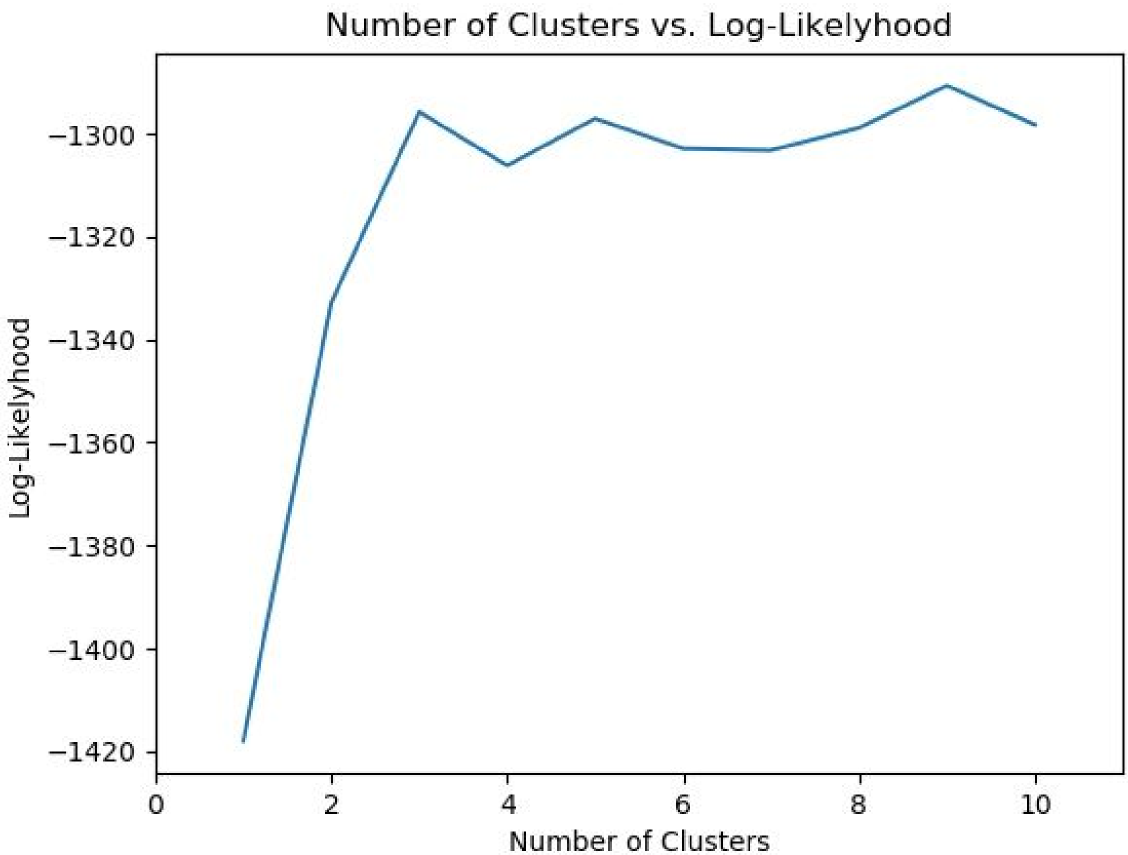


Figure 2.1: Log-Likelihood vs Number of Clusters

- From the Fig. (2.1), it can be seen that after *k* = 3 (i.e. number of clusters = 3), Log-Likelihood doesn't increase by much. Thus, we can say that the process that generated this dataset is likely to have 3 Clusters.

Problem 3

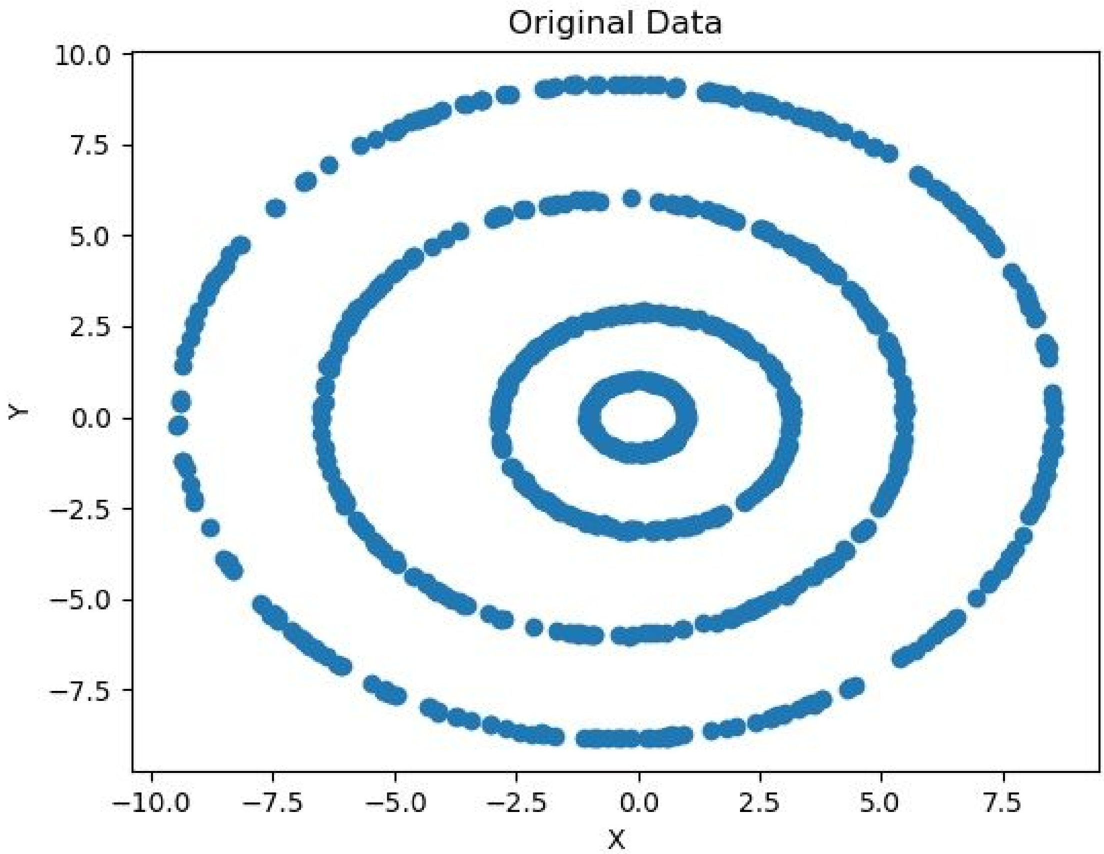


Figure 3.1: Plot of Original Data

Part 1 : Covariance Matrix is

$$\begin{matrix} 7.3831 & 0.4044 \\ 0.4044 & 8.4277 \end{matrix}$$

and Eigenvalues of Covariance Matrix are 8.5659 and 7.2448 , contribution of PC1 and PC2 are given below. Contribution computed by the formulae $\lambda i = \lambda i / \sum_{i=1}^{n} \lambda i$.

EigenValues	Contribution(%)
λ1(8.5659)	54.1780
λ2(7.2448)	45.8220

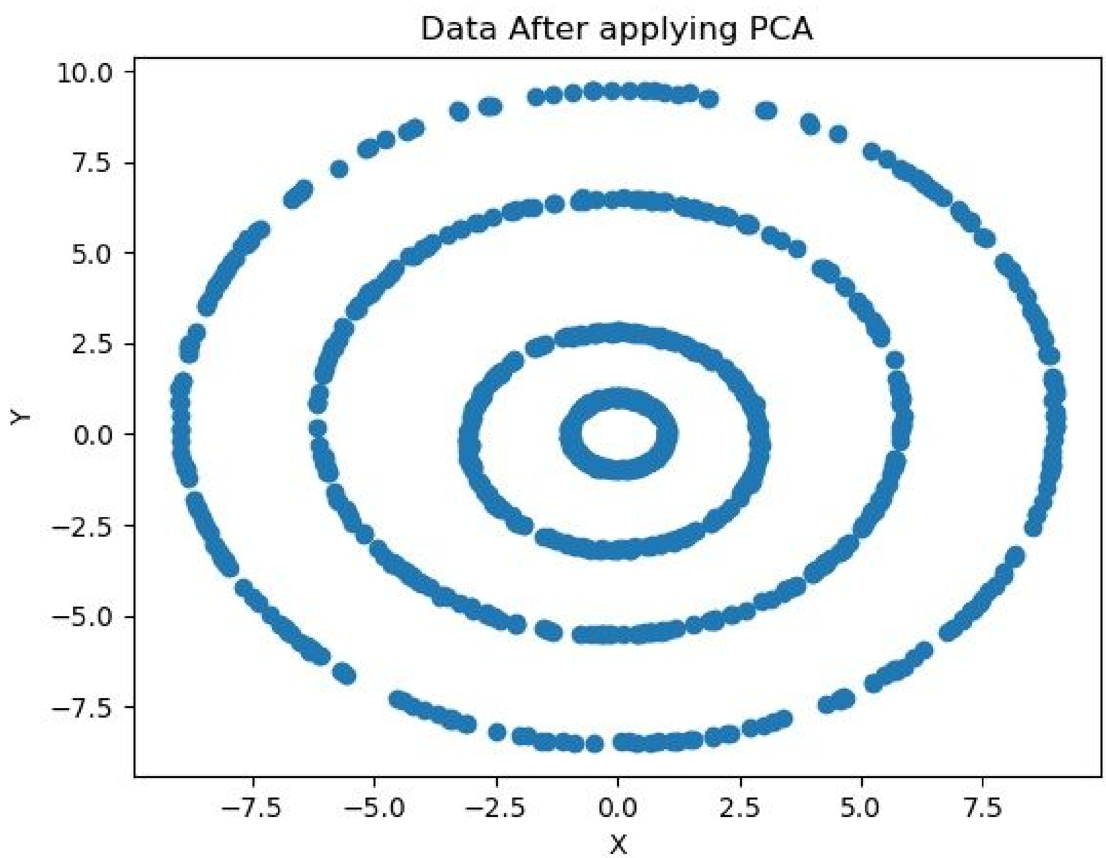


Figure 3.2 : Plot Of Data After PCA

Part 2(A):

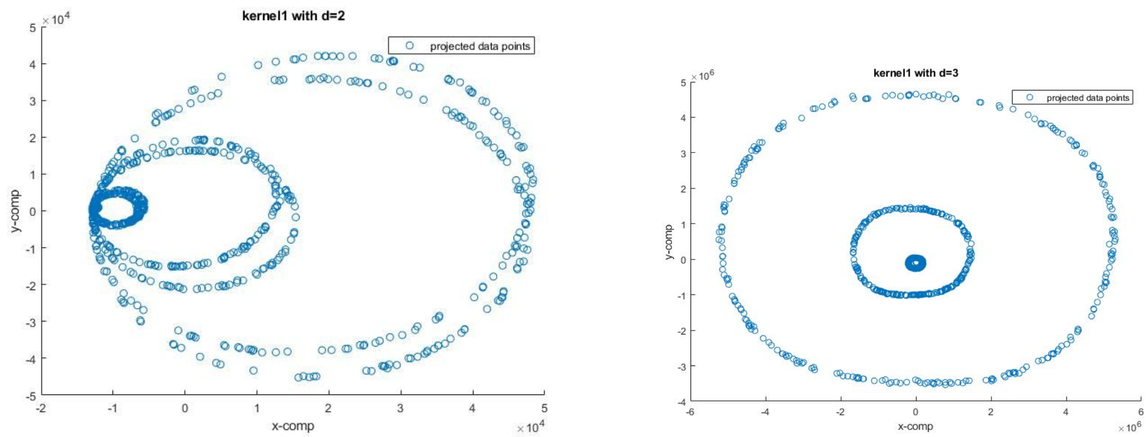
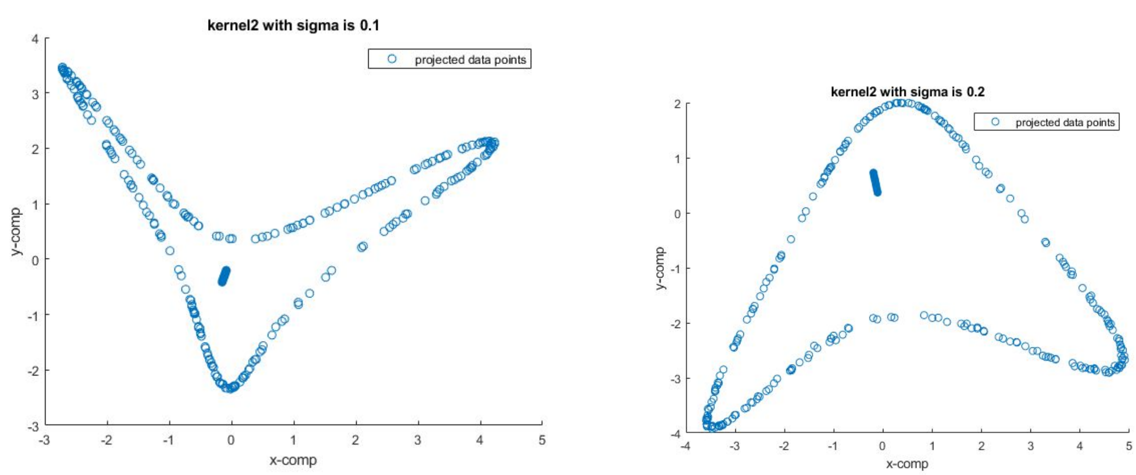
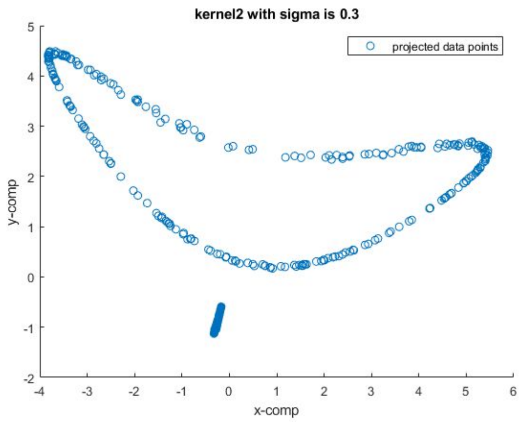
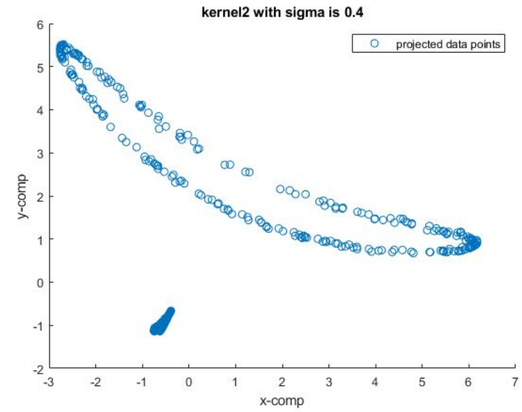


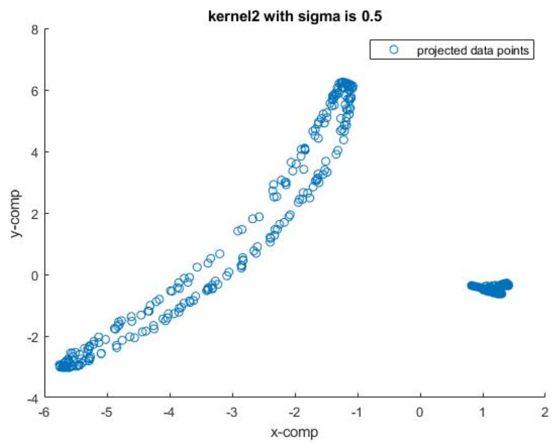
Figure 3.3 : Projection Of Data Points After Applying Kernel(1) With d =2 ,3

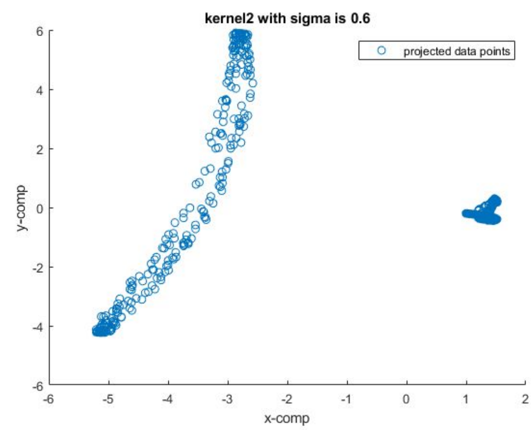
Part 2(B):




kernel2 with sigma is 0.3
projected data points
5
4
3
2
1
0
-1
-2
y-comp
-4 -3 -2 -1 0 1 2 3 4 5 6
x-comp


kernel2 with sigma is 0.4
projected data points
6
5
4
3
2
1
0
-1
-2
y-comp
-3 -2 -1 0 1 2 3 4 5 6 7
x-comp


kernel2 with sigma is 0.5
projected data points
8
6
4
2
0
-2
-4
y-comp
-6 -5 -4 -3 -2 -1 0 1 2
x-comp


kernel2 with sigma is 0.6
projected data points
6
4
2
0
-2
-4
-6
y-comp
-6 -5 -4 -3 -2 -1 0 1 2
x-comp

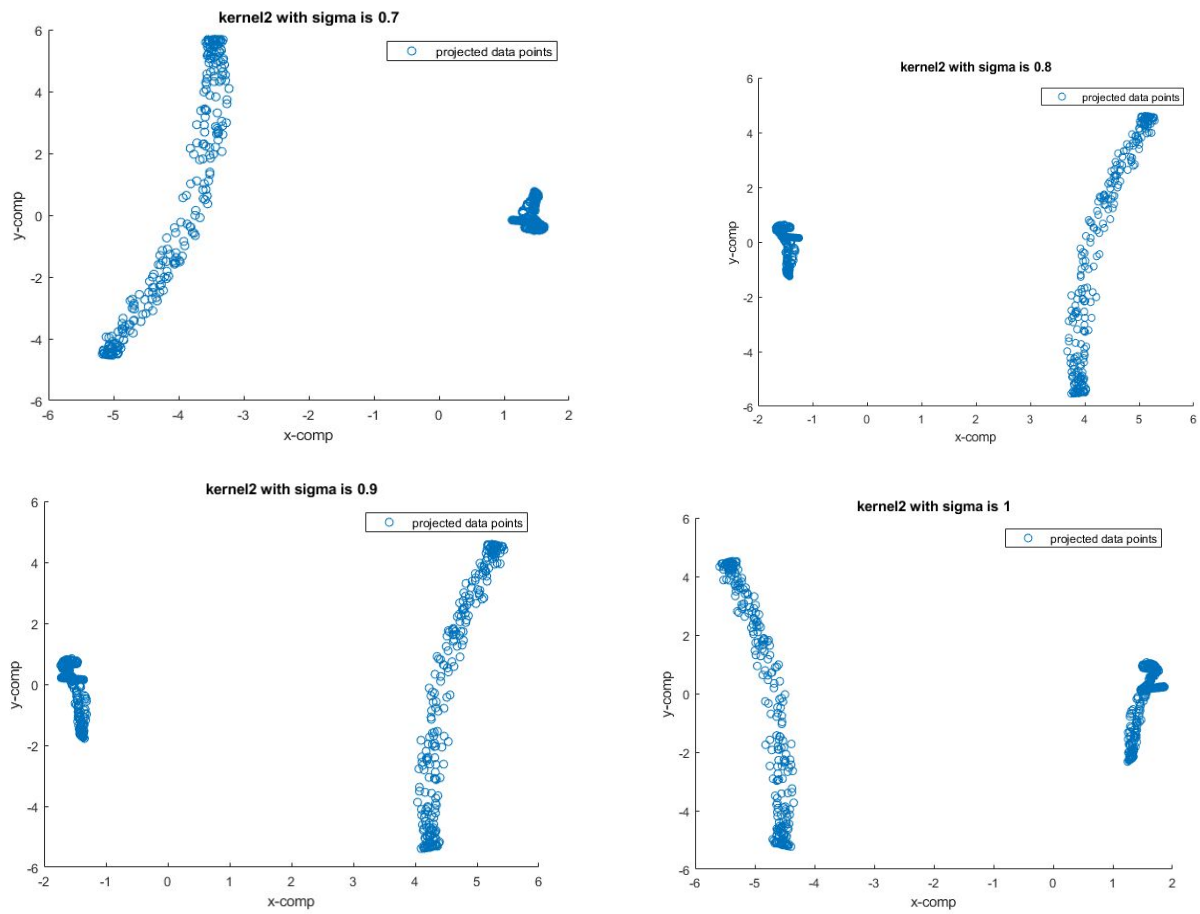


Figure 3.4 : Projection Of Data Points After Applying Kernel(2) With σ = {0.1, 0.2 , . . . , 1}

Part 3 :

Kernel B is best suited for given dataset because for σ >=0.3 one can linearly separate the data, So it will be easy to classify.

Note: Tools Used: Matlab, Python [Numpy (Array, Zeros, Ones, Sum), Matplotlib]