**Introduction to ML (CS771), Autumn 2023**
**Indian Institute of Technology Kanpur**
**Homework Assignment Number 2**

*Student Name:* Ogirala Deeven Kumar
*Roll Number:* 210681
*Date:* November 17, 2023

# QUESTION 1

Standard k-means objective function is given as,

$$\mathcal{L}(\mathbf{X}, \mathbf{Z}, \boldsymbol{\mu})) = \sum_{n=1}^{N}\sum_{k=1}^{K} \boldsymbol{z_{nk}}\|\boldsymbol{x_n} - \boldsymbol{\mu}_k\|^2$$

**SGD k-means:**

**Step - 1:**
To make it online by taking a random example $\mathbf{x}_n$ at a time, and then assigning $\mathbf{x}_n$ greedily to the best cluster, using **ALT-OPT** technique,
We'll fix $\boldsymbol{\mu} = \hat{\boldsymbol{\mu}}$ and solve for $\boldsymbol{z}_n$

$$\begin{aligned}\hat{\boldsymbol{z}}_n &= \arg \underset{\hat{\boldsymbol{z}}_n}{min} \sum_{k=1}^{K} \boldsymbol{z_{nk}}\|\boldsymbol{x_n} - \boldsymbol{\mu}_k\|^2 \\ &= \arg \underset{\hat{\boldsymbol{z}}_n}{min} \boldsymbol{z_{nk}}\|\boldsymbol{x_n} - \boldsymbol{\mu}_{\boldsymbol{z}_n}\|^2\end{aligned}$$

To do the step-1 we need to assign a cluster to $\boldsymbol{x}_n$ using the above equation for each example $\{\boldsymbol{x}_n\}_{n=1}^{N}$.

**Step - 2:**
To update the cluster means, solving for $\boldsymbol{\mu}$ using SGD on the objective function by fixing $\boldsymbol{z} = \hat{\boldsymbol{z}}$

$$\begin{aligned}\hat{\boldsymbol{\mu}} &= \arg \underset{\mu}{min} \mathcal{L}(\boldsymbol{X}, \hat{\boldsymbol{Z}}, \mu) \\ &= \arg \underset{\mu}{min} \{\sum_{n=1}^{N} \sum_{n:\hat{\boldsymbol{z}}_n=k}^{K} \|\boldsymbol{x_n} - \boldsymbol{\mu}_k\|^2\} \\ \hat{\boldsymbol{\mu}}_k &= \arg \underset{\mu_k}{min} \{\sum_{n:\hat{\boldsymbol{z}}_n=k}^{K} \|\boldsymbol{x_n} - \boldsymbol{\mu}_k\|^2\}\end{aligned}$$

At any iteration $\mathbf{t}$, we choose an example $\boldsymbol{x}_n$ uniformly randomly and approximate $\mathbf{g}$ as

$$\boldsymbol{g} \approx \boldsymbol{g}_n = \frac{\partial}{\partial \mu_k}(\boldsymbol{x_n} - \boldsymbol{\mu}_k\|^2) = -2(\boldsymbol{x_n} - \boldsymbol{\mu}_k)$$

Mean can be updated as, $\mu_k^{(t+1)} = \boldsymbol{\mu}_k^{(t)} - \eta \boldsymbol{g}^{(t)}$
$\implies \mu_k^{(t+1)} = \boldsymbol{\mu}_k^{(t)} + 2\eta(\boldsymbol{x}_n^{(t)} - \boldsymbol{\mu}_k^{(t)})$

Step size can be $\eta \propto \frac{1}{N_k}$, where $N_k$ is the number of data points in $k^{th}$ cluster so that the updated mean would also be in the ratio of sum of features of every data point to the total number of data points in the cluster.

**Introduction to ML (CS771), Autumn 2023**
**Indian Institute of Technology Kanpur**
**Homework Assignment Number 2**

*Student Name:* Ogirala Deeven Kumar
*Roll Number:* 210681
*Date:* November 17, 2023

# QUESTION 2

To project the inputs into one dimension such that the distance between the means of the two classes is maximized and the inputs within each class are as close to each other as possible, you can use the following objective function based on Fisher's Linear Discriminant Analysis (LDA):

$$J(w) = \frac{\mathbf{w}^T(\mu_1 - \mu_2)(\mu_1 - \mu_2)^T\mathbf{w}}{\mathbf{w}^T(S_1 + S_2)\mathbf{w}}$$

where,
w is the projection direction vector (a unit vector). $\mu_1$ and $\mu_2$ are the means of the data points in class +1 and -1, respectively. $S_1$ and $S_2$ are the scatter matrices for class +1 and -1, respectively. The scatter matrix for each class is given by $S_i = \sum_{n=1} N_i(\boldsymbol{x}_n - \boldsymbol{\mu}_i)(\boldsymbol{x}_n - \boldsymbol{\mu}_i)^T$, where $\mathbf{N}_i$ is the number of data points in class i, $x_n$ is a data point, and $\mu_i$ is the mean of class i.

The objective is to maximize the ratio of the between-class scatter to the within-class scatter. This encourages the projection direction that maximizes the separation between the means of the two classes while minimizing the dispersion within each class.

**Introduction to ML (CS771), Autumn 2023**
**Indian Institute of Technology Kanpur**
**Homework Assignment Number 2**

# QUESTION 3

*Student Name:* Ogirala Deeven Kumar
*Roll Number:* 210681
*Date:* November 17, 2023

Assume a eigen vector $\mathbf{v}$ of matrix $\mathbf{S} = \frac{1}{N}\mathbf{X}\mathbf{X}^T$ which will satisfy, $\mathbf{S}\mathbf{v} = \lambda\mathbf{v}$, where $\lambda$ is the eigen value.

$$\Rightarrow \frac{1}{N}(\mathbf{X}\mathbf{X}^T)\mathbf{v} = \lambda\mathbf{v}$$
$$\Rightarrow \frac{1}{N}(\mathbf{X}^T\mathbf{X})(\mathbf{X}^T\mathbf{v}) = \lambda(\mathbf{X}_T\mathbf{v})$$

let $\mathbf{u} = \mathbf{X}^T\mathbf{v} \Rightarrow \frac{1}{N}(\mathbf{X}\mathbf{X}^T)\mathbf{u} = \lambda\mathbf{u}$. Therefore $\mathbf{u} = \mathbf{X}^T\mathbf{v}$ is the eigen vector for $\frac{1}{N}(\mathbf{X}\mathbf{X}^T)$
In normal case to compute k eigen vector for $\frac{1}{N}(\mathbf{X}^T\mathbf{X})$ complexity is $O(KD^2)$, but the complexity here is $O(KN^2)$ for decomposition of $\frac{1}{N}(\mathbf{X}\mathbf{X}^T) + O(KND)$ for matrix multiplication.

Therefore, the complexity is $O(KND)$, cause $D > N$ .

**Introduction to ML (CS771), Autumn 2023**
**Indian Institute of Technology Kanpur**
**Homework Assignment Number 2**

*Student Name:* Ogirala Deeven Kumar
*Roll Number:* 210681
*Date:* November 17, 2023

# QUESTION 4

**Part-1:** A standard linear model will only works when we have to regress a linear curve whereas this model can be a combination of K different linear curves,what this model does is that at first, clustering the data on k different linear curves and then the prediction are made for y, which also helps in the reduction of outliers in a linear curve as the outliers may get separate out due to clustering.

**Part-2:**
Latent variable model is,

$$\begin{aligned}
p(z_n = k|y_n, 0) &= \frac{p(z_n = k)p(y_n|z_n = k, 0)}{\sum_{l=1}^{K} p(z_n = l)p(y_n|z_n = l, 0)} \\
p(y_n, z_n|0) &= p(y_n|z_n, 0)p(z_n|0) \\
\text{where, } p(z_n = k) &= \pi_k \\
p(y_n|z_n, 0) &= N(w_{z_n}^T x_n, \beta^{-1})
\end{aligned}$$

**ALT-OPT Algorithm:**
Step-1: To find the best $z_n$

$$\begin{aligned}
z_n &= \arg \underset{z_n}{min} \frac{\pi_k N(w_{z_n}^T x_n, \beta^{-1})}{\sum_{l=1}^{K} \pi_l N(w_l^T x_n, \beta^{-1})} \\
\Rightarrow z_n &= \arg \underset{z_n}{min} \frac{\pi_k \exp(\frac{-\beta}{2}(y_n - w_{z_n}^T x_n)^2)}{\sum_{l=1}^{K} \pi_k \exp(\frac{-\beta}{2}(y_n - w_l^T x_n)^2)}
\end{aligned}$$

Step-2: re-estimating the parameters

$$\begin{aligned}
N_k &= \sum_{n=1}^{N} z_{nk} \\
w_k &= (\boldsymbol{X}_k^T \boldsymbol{X}_k)^{-1} \boldsymbol{X}_k^T y_k \\
\pi_k &= \frac{N_k}{N}
\end{aligned}$$

Here $\mathbf{X}_k$ is $N_k \times D$ matrix containing training sets which is clustered in class k. $y_n$ is $N_k \times 1$ vector containing training set labels which is clustered in class k.
if $\pi_k = \frac{1}{K}$ then,

$$z_n = \arg \underset{z_n}{min} \frac{\exp(\frac{-\beta}{2}(y_n - w_{z_n}^T x_n)^2)}{\sum_{l=1}^{K} \exp(\frac{-\beta}{2}(y_n - w_l^T x_n)^2)}$$

this is equivalent to multi output logistic regression.

**Introduction to ML (CS771), Autumn 2023**
**Indian Institute of Technology Kanpur**
**Homework Assignment Number 2**

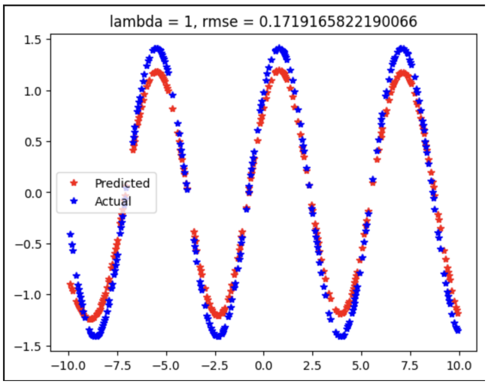
*Student Name:* Ogirala Deeven Kumar
*Roll Number:* 210681
*Date:* November 17, 2023

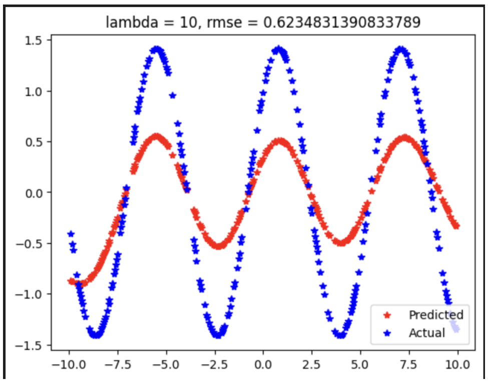
# QUESTION 5

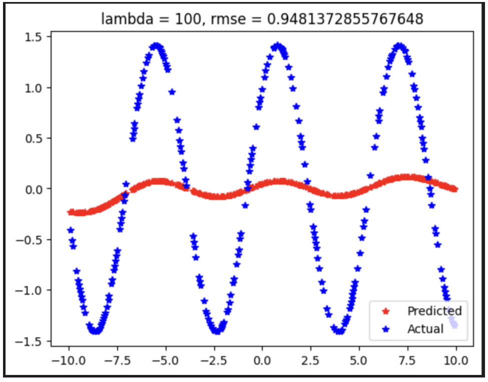
**Part-1:**

**For kernel ridge regression:** The increase in the regularization hyper parameter appears to coincide with a rise in error. This alignment might be due to the inherent similarity between the training and test data sets, derived from a similar sine curve without substantial outliers. Lower regularization facilitates a closer fit to the training data, indirectly improving performance on the test data.

**Plots:**

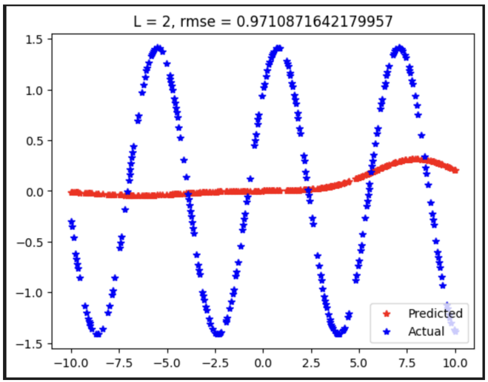


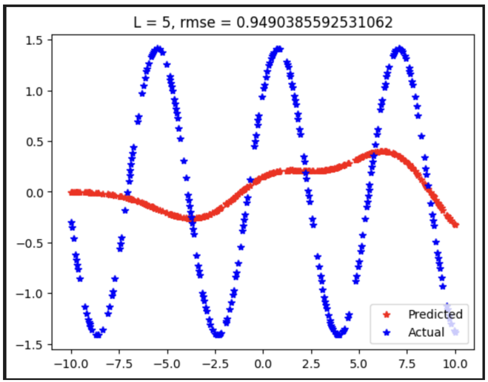


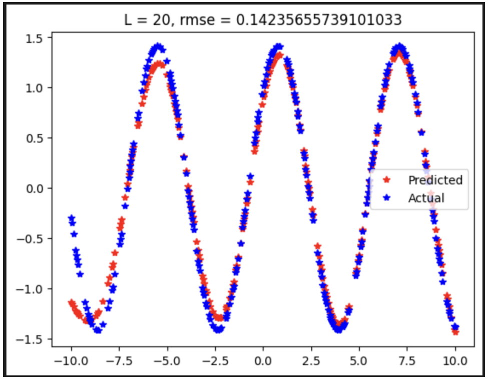


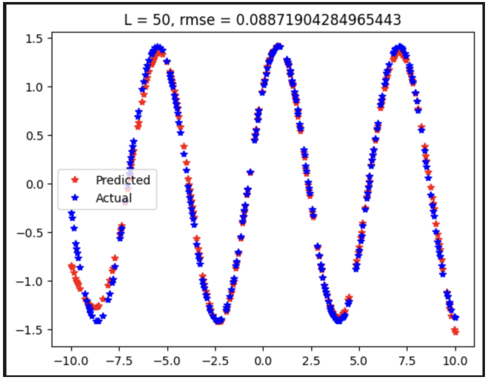
**For landmark-ridge:** A lower value for the hyper parameter L results in higher prediction errors, likely stemming from a reduced number of feature points considered. Interestingly, L=50 seems to strike a good balance, as further increasing L to 100 marginally impacts the RMSE by merely 0.02.

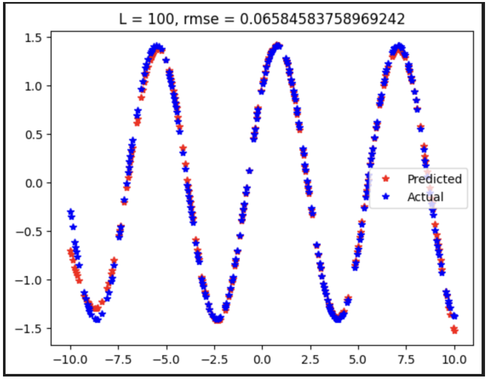
## Plots:







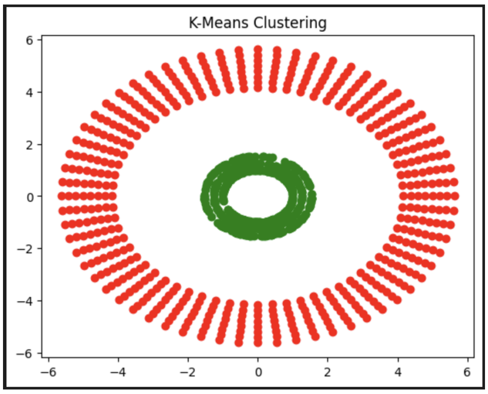




## Part-2:

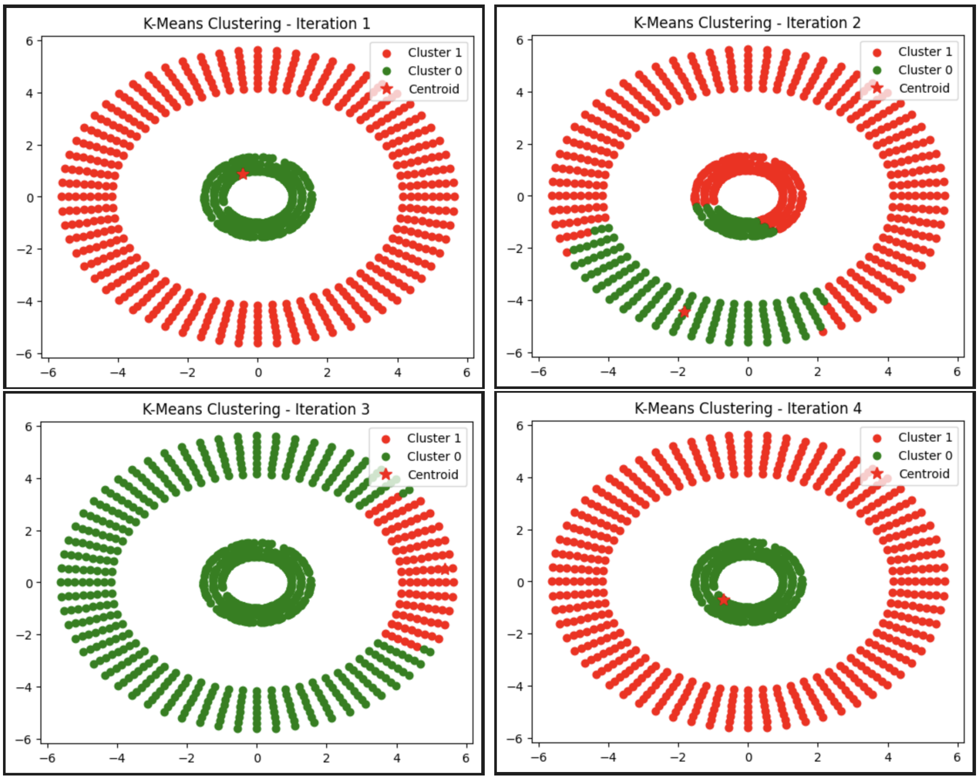
**Using Hand-crafted Features:** Upon plotting the data, it becomes evident that the clusters exhibit a radial distribution around the origin. Consequently, a feature transformation based on the distance from the origin was employed in the handcrafted part.

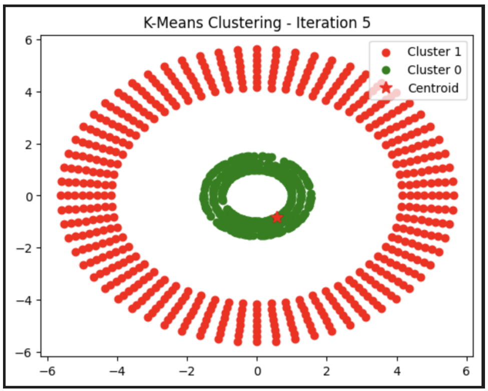
**Plots:**

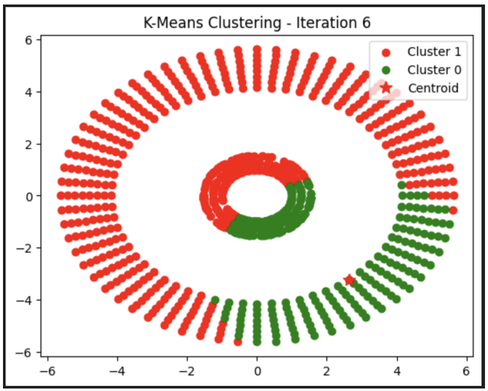


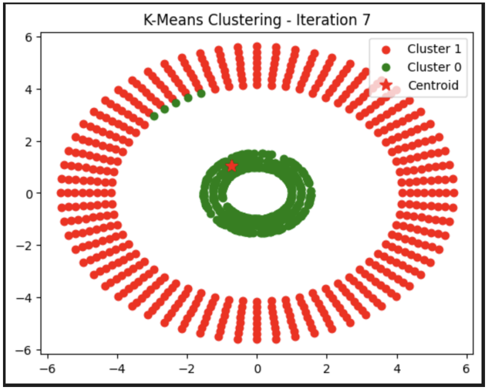
**Using Kernels:** The data's radial distribution around the origin lends itself well to clustering through a feature transformation utilizing point-to-origin distance. The efficacy of a landmark in clustering is influenced by its proximity to the origin; a closer landmark tends to cluster the data effectively, while a farther one may struggle due to the radial nature of the dataset around the origin.

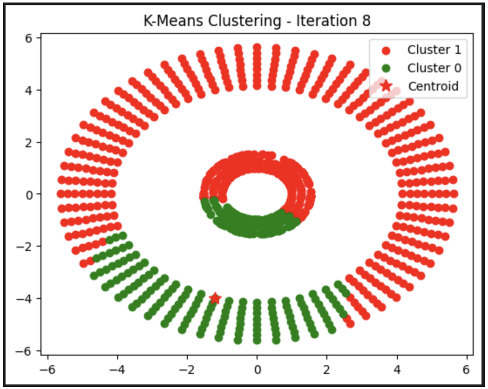
**Plots:**

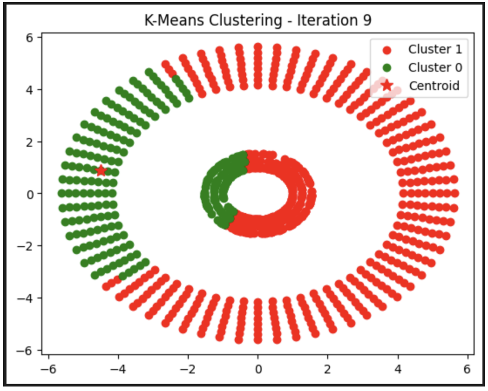


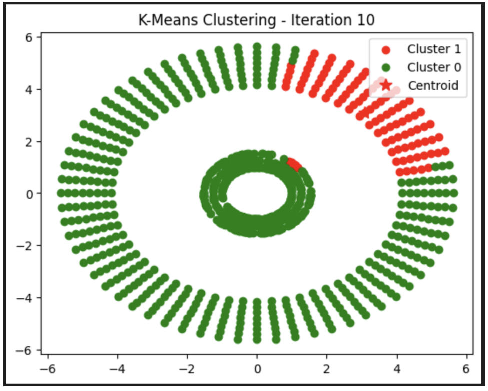

K-Means Clustering - Iteration 5
Cluster 1
Cluster 0
Centroid
6
4
2
0
−2
−4
−6
−6
−4
−2
0
2
4
6


K-Means Clustering - Iteration 6
Cluster 1
Cluster 0
Centroid
6
4
2
0
−2
−4
−6
−6
−4
−2
0
2
4
6


K-Means Clustering - Iteration 7
Cluster 1
Cluster 0
Centroid
6
4
2
0
−2
−4
−6
−6
−4
−2
0
2
4
6

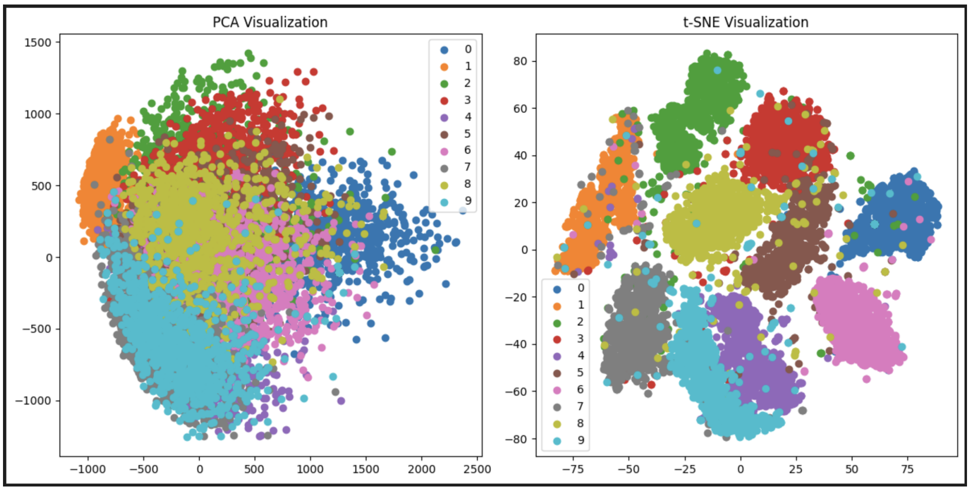

K-Means Clustering - Iteration 8
Cluster 1
Cluster 0
Centroid
6
4
2
0
−2
−4
−6
−6
−4
−2
0
2
4
6


K-Means Clustering - Iteration 9
Cluster 1
Cluster 0
Centroid
6
4
2
0
−2
−4
−6
−6
−4
−2
0
2
4
6


K-Means Clustering - Iteration 10
Cluster 1
Cluster 0
Centroid
6
4
2
0
−2
−4
−6
−6
−4
−2
0
2
4
6

**Part-3:**

PCA plots may show clustering of data points but might not capture intricate local structures or groupings present in the original data. t-SNE plots often reveal finer structures and clusters compared to PCA plots. Points belonging to the same class tend to form tighter clusters, and it's better at preserving the local neighborhood relationships. In t-SNE, distances between clusters might not have any meaningful interpretation since the focus is on local relationships. PCA captures the overall variance and some large-scale relationships, t-SNE emphasizes local structures and finer patterns


PCA Visualization
t-SNE Visualization