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

Student Name: Divyansh
Roll Number: 210355
Date: November 16, 2023

QUESTION 1

Part 1: Assigning the data point x_n greedily to the best cluster

Selecting a x_n randomly from the data examples. Now, I will assign x_n to the closest cluster k^n by based on the distances between x_n and all cluster centers μ_k where $k \in \{1, 2, ...K\}$ assuming K is the total number of clusters.

Predicted cluster for point x_n be $Cluster_k$

$$Cluster_k = \{n : k = argmin_k \left|\mathbf{x}_n - \mu_n\right|^2\}$$

Part 2: The SGD-based cluster mean update equation

Let N_k be the count of points in cluster k, we increase the N_k by 1 when the random point x_n gets assigned to the k^{th} cluster.

Now our loss function objective was:

$$\mathcal{L} = \sum_{n=1}^{n_k} z_{nk} \left|\mathbf{x}_n - \mu_n\right|^2$$

Since, $\mathbf{x}_n$ belongs to cluster k so, $z_{nk} = 1$. So, the modified loss funciton is:

$$\mathcal{L} = \sum_{n=1}^{n_k} \left|\mathbf{x}_n - \mu_n\right|^2$$

Minimizing this loss function with respect to μ_k we have the gradient:

$$g_{\mu_k} = \frac{\partial \mathcal{L}}{\partial \mu_k} = -\sum_{n=1}^{n_k} 2(\mathbf{x}_n - \mu_k) = -2\big(\sum_{k=1}^{n_k} x_n - \sum_{k=1}^{n_k} \mu_k\big)$$

$$\implies g_{\mu_k} = -2\big(\mu_k(n_k - 1) + \mathbf{x}_n - n_k\mu_k\big) = -2\big(\mathbf{x}_n - \mu_k\big)$$

Using the above gradient, the SGD equation of updation turns out:

$$\mu_k^t = \mu^{t-1} - \alpha g_{\mu_k}^{t-1}$$

(where alpha is the learning rate)

$$\implies \mu_k^t = \mu^{t-1} + 2\alpha\big(\mathbf{x}_n - \mu_k^{t-1}\big)$$

The update equation justifies because the new mean should be dependent on the mean of the points in that cluster and also the the change in the mean is being controlled by a hyper parameter α.

Suitable choice of learning rate will be $\frac{1}{n_k}$ because that makes the update equation same as obtaining the new average of all data points in the k^{th} cluster to which the random choosen example belonged. It also satisfies the property of a good learning rate since, as the number of points getting assigned to cluster increases the learning keeps tending towards zero.

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

Student Name: Divyansh
Roll Number: 210355
Date: November 16, 2023

QUESTION 2

Taking inspiration from **Fisher's criterion** [ref] for this dimensionality reduction and classification process

Let's define two matrices, first one is between-class scatter matrix ($\mathbf{S}_B$) and other one is within-class scatter matrix($\mathbf{S}_W$) which is defined as follows:

$$\mathbf{S}_B = (\mu_+ - \mu_-)(\mu_+ - \mu_-)^T$$

Where μ_+ and μ_- are the means of the inputs from the positive and negative classes, respectively. This measures the spread between different classes. And,

$$\mathbf{S}_W = \sum_{n:y_n=+1} (x_n - \mu_+)(x_n - \mu_+)^T + \sum_{n:y_n=-1} (x_n - \mu_-)(x_n - \mu_-)^T$$

This measures spread within each class. Now, using this the obective function created is

$$\mathcal{L}(\mathbf{w}) = \frac{\mathbf{w}^T S_B \mathbf{w}}{\mathbf{w}^T S_W \mathbf{w}}$$

Justification: Maximizing this ratio ensures that the means of different classes are as far apart as possible relative to the spread within each class. The larger the value of $\mathcal{L}(\mathbf{w})$, the better the separation.

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

Student Name: Divyansh
Roll Number: 210355
Date: November 16, 2023

QUESTION 3

Let $\mathbf{v}$ be an eigenvector of the covariance matrix $\mathbf{S} = \frac{1}{N}\mathbf{X}\mathbf{X}^T$, then we have the following equation:

$$\frac{1}{N}\mathbf{X}\mathbf{X}^T\mathbf{v} = \lambda\mathbf{v}$$

Multiplying the equation on the left by $\mathbf{X}^T$, we get the following:

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

This implies that $\mathbf{u} = \mathbf{X}^T\mathbf{v}$ is an eigenvector of $\mathbf{X}^T\mathbf{X}$. So for every eigenvector $\mathbf{v}$ of $\frac{1}{N}\mathbf{X}\mathbf{X}^T$, we have $\mathbf{u} = \mathbf{X}^T\mathbf{v}$ which is an eigenvector of $\frac{1}{N}\mathbf{X}^T\mathbf{X}$.

This means that we need to do eigendecomposition of an $N \times N$ matrix instead of a $D \times D$ matrix. Since $D > N$, this method is more computationally efficient.

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

Student Name: Divyansh
Roll Number: 210355
Date: November 16, 2023

QUESTION 4

Part 1: Brief explanation of discussed model

Unlike a standard linear model that assumes a single linear relationship between the input and output variables, this model allows for K different linear relationships. It first assigns each data point to one of the K linear clusters, and then predicts the output variable based on the cluster's linear equation. This can reduce the impact of outliers that do not fit well in a single linear curve, as they may belong to a different cluster.

Part 2: ALT-OPT algorithm

1. Initialize Θ as $\hat{\Theta}$
2. For $n = \{1...N\}$ we will find the best $\mathbf{z}_n$ as follows:

$$\hat{\mathbf{z}_n} = \arg\max_k \; p(\mathbf{y}_n | z_n = k, \hat{\Theta})$$

$$\hat{\mathbf{z}_n} = \arg\max_k \; p(z_n = k|\hat{\Theta}) p(\mathbf{y}_n | z_n = k, \hat{Theta})$$

$$\hat{\mathbf{z}_n} = \arg\max_k \; \pi_k \mathcal{N}(\mathbf{w}_k^T \mathbf{x}_n, \beta^{-1})$$

3. Given $\mathbf{Z} = \{\hat{z_1}, ...\hat{z_n}\}$, we will restimate Θ using **MLE**.

$$\hat{\Theta} = \arg\max_\Theta \; logp(\mathbf{Y}, \hat{\mathbf{Z}}|\Theta)$$

After simplification we got:

$$\hat{\Theta} = \arg\max_\Theta \sum_{n=1}^{N} \sum_{k=1}^{K} \hat{z_{nk}} [log(\pi_k) + log\mathcal{N}(\mathbf{w}_k^T \mathbf{x}_n, \beta^{-1})]$$

The update for π_k will be same as in the case of generative classification model. To find the update for $\mathbf{w}_k$ we can write the expression like:

$$\mathbf{w}_k = \arg\max_{\mathbf{w}_k} \sum_{n:\hat{z_{nk}}=1} log\mathcal{N}(\mathbf{w}_k^T \mathbf{x}_n, \beta^{-1})$$

If we compare this with the objective function for linear regression problem, the only difference is that only a subset of the examples contribute, hence the Update equation is

$$\hat{\pi_k} = \frac{N_k}{N}$$

$$\boxed{\hat{\mathbf{w}_k} = (\sum_{n:\hat{z_{nk}}=1} \mathbf{x}_n \mathbf{x}_n^T)^{-1} \sum_{n:\hat{z_{nk}}=1} y_n \mathbf{x}_n}$$

4. Go to step 2 if not yet converged

if $\pi_k = \frac{1}{k}$ we will update z_n as following:
$\hat{z_n} = \arg\max_k \mathcal{N}(\mathbf{w}_k^T \mathbf{x}_n, \beta^{-1})$
This update chooses the $\mathbf{w}_k$ which results in highest porbability or least square error for the data point $(\mathbf{x}_n, y_n)$

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

Student Name: Divyansh
Roll Number: 210355
Date: November 16, 2023

QUESTION
5

1 Part 1

Subpart (i)

1. **Kernel ridge regression**

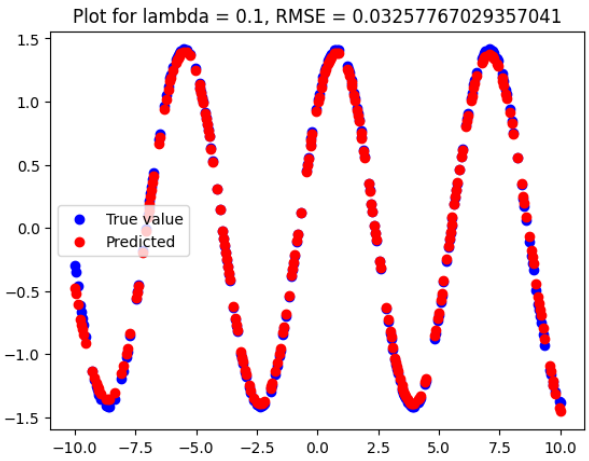

Figure 1: Lamda = 0.1, RMSE = 0.03257767029357041

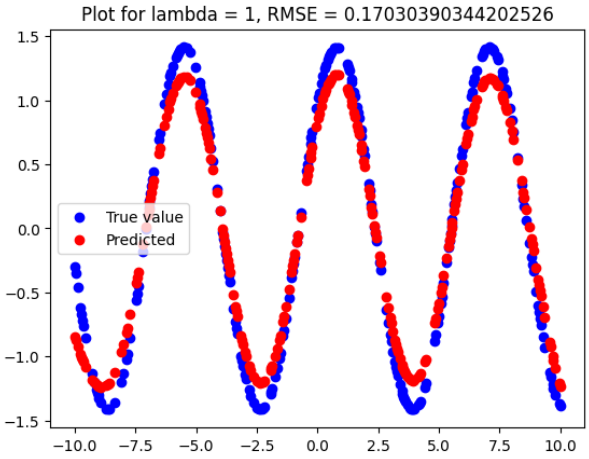

Figure 2: Lamda = 1, RMSE = 0.17030390344202526

Observation: The RMSE score keepings decreasing as we increase the values of λ which means that less regularization is giving better results in case of kernel ridge regression.

2. **Landmark-ridge regression**

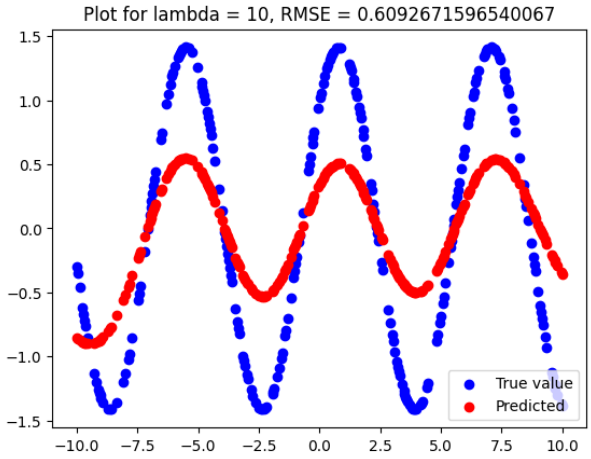

Figure 3: Lamda = 10 - RMSE = 0.6092671596540067

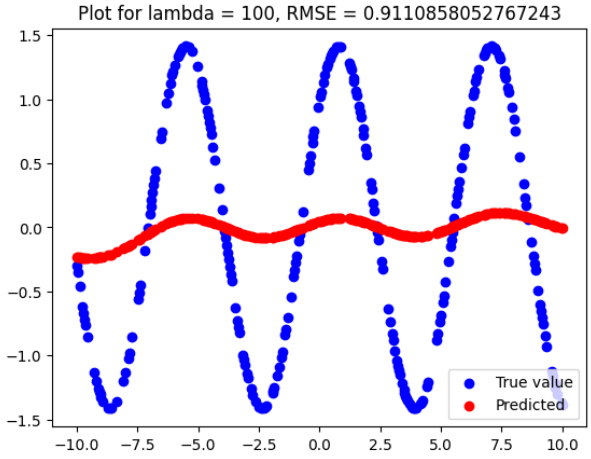

Figure 4: Lamda = 100, RMSE = 0.9110858052767243

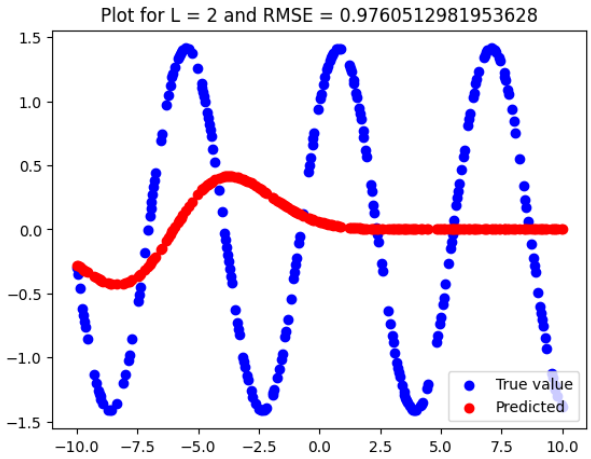

Figure 5: L = 2 and RMSE = 0.9760512981953628

Obervation: unlike the case of kernel ridge, in the case of landmark regression the score increases as we increase the number of landmarks in the task. The RMSE of each case is

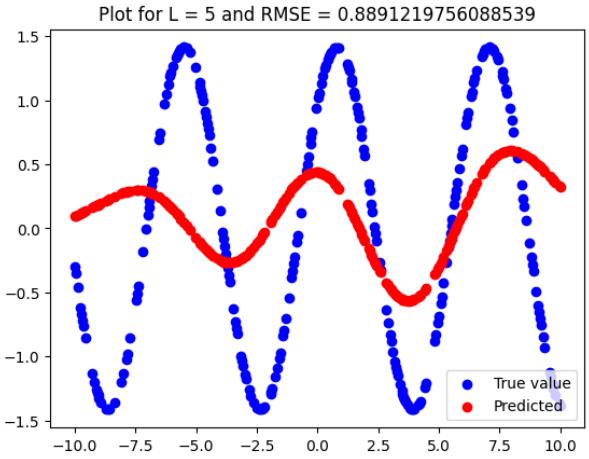

Figure 6: L = 5 and RMSE = 0.8891219756

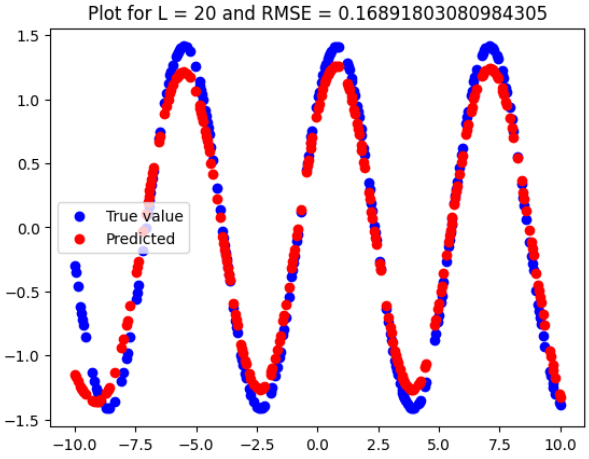

Figure 7: L = 20 and RMSE = 0.16891803080984305

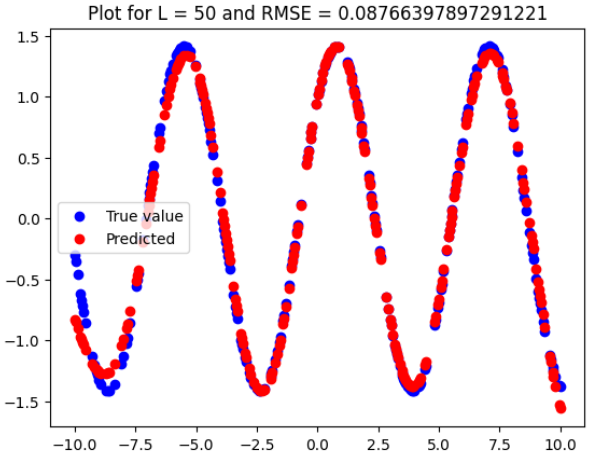

Figure 8: L = 50 and RMSE = 0.08766397897291221

mentioned in the image(caption) itself. The best result came across the L = 100.

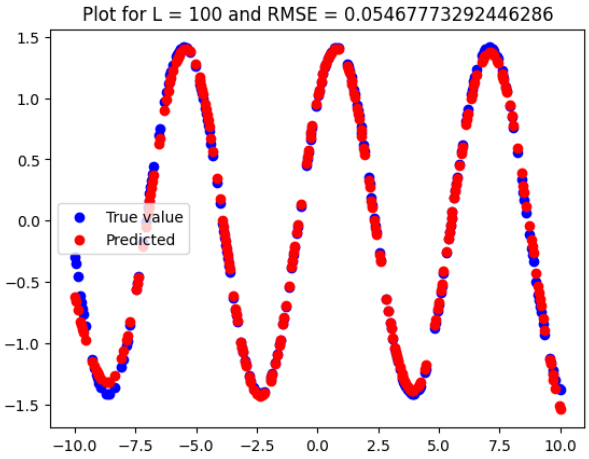

Figure 9: L = 100 and RMSE = 0.05467773292446286

2 Part 2

1. **Using Handcrafter Features**

   For Handcrafted features I used the distance of the points from origin and that gave a single dimensional feature for each point in the dataset.

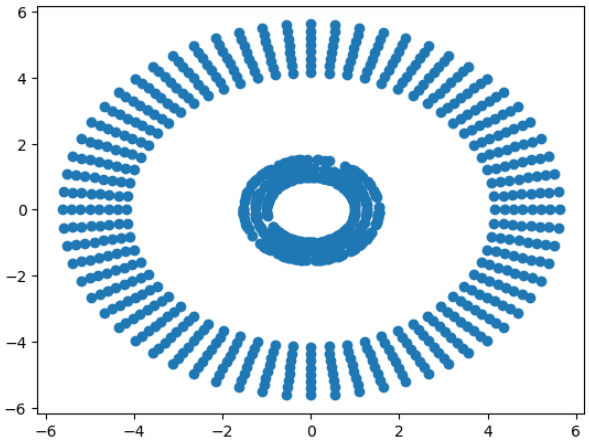

Figure 10: Original Data

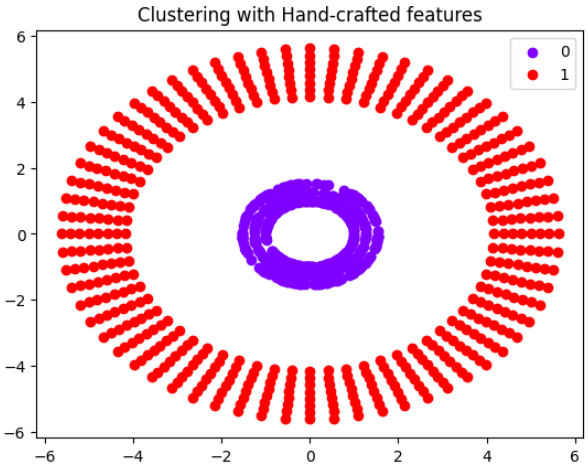

Figure 11: CLustering with Hand-crafted features

2. **Using Kernels with a Landmark**

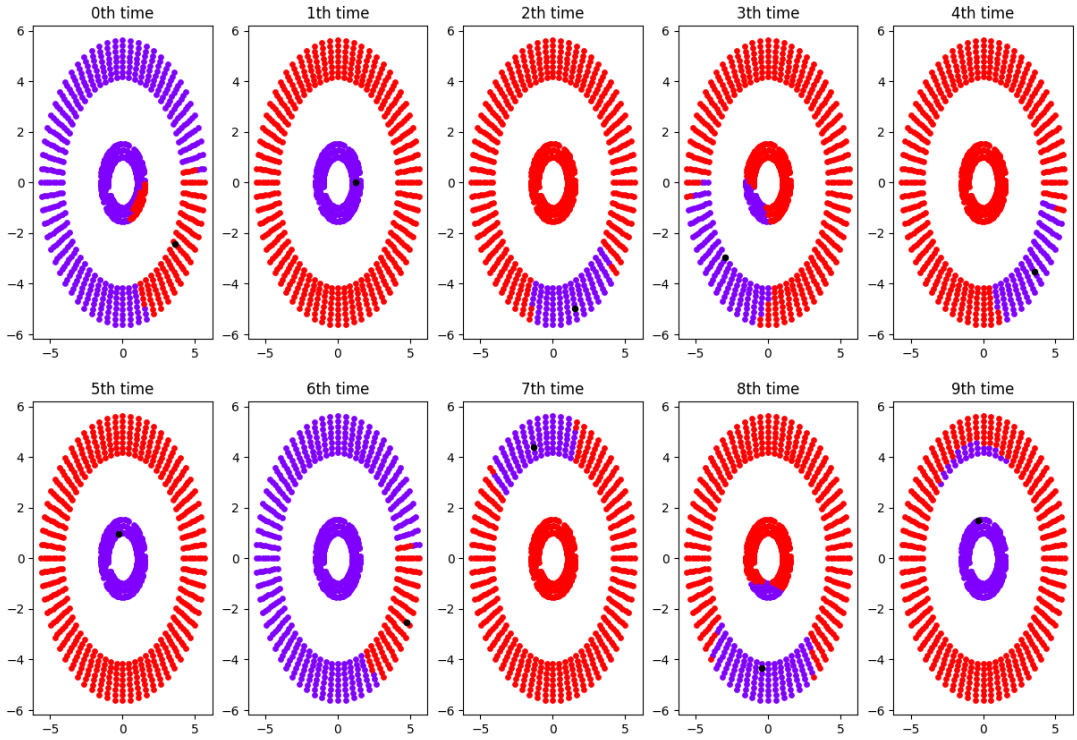

Figure 12: Using single landmark(shown in black colour point)

Justification: In the handcrafted features part, the distance from the oirigin was very useful feature since it clearly divided the dataset into two valid clusters. In case of landmark whenever the randomly selected point lies near the origin the clustering takes the similar clustering like the hand-crafted one while in opposite case the clustering turns out be bad.

3 Part 3

Visualization of Images on 2D Plot

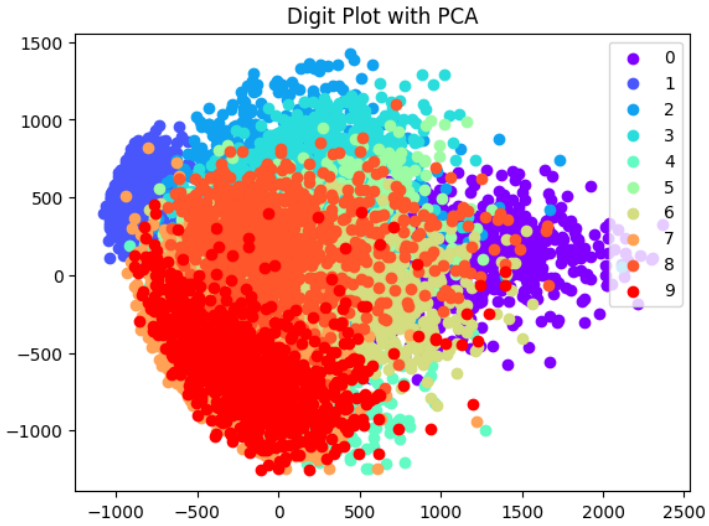

Figure 13: Plotting using PCA

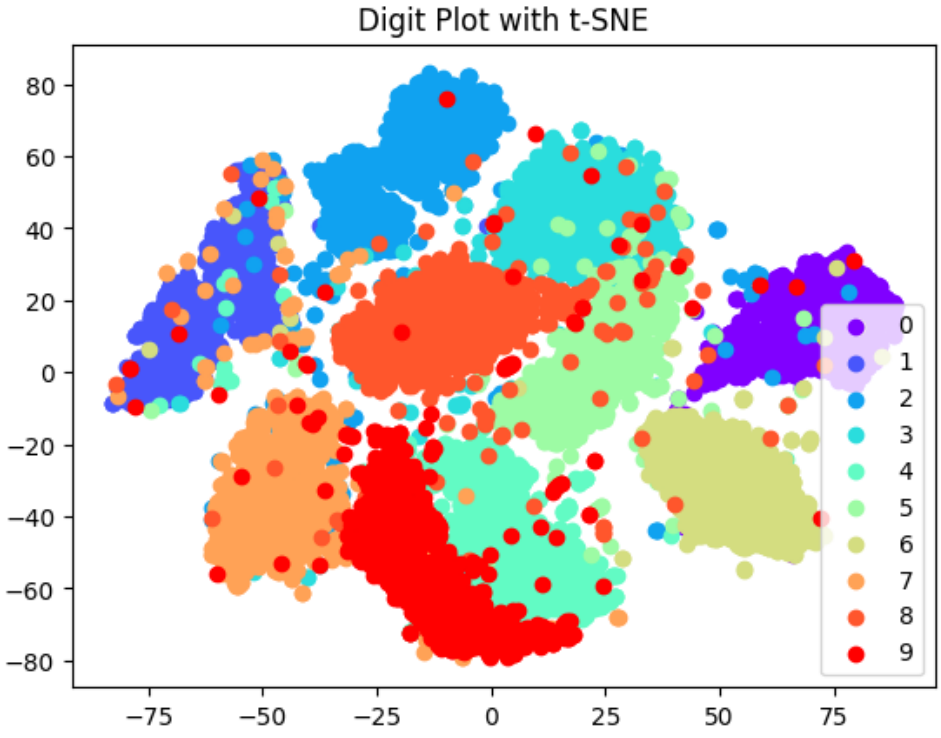

Figure 14: Plotting using t-SNE

Difference: The t-SNE method was clearly better amongs the two method tried since it was able to distinguish the images of different digits in much more clearer way with less overlap while the PCA method made clusters miserably with most of the part overlapping in th plot.