Machine Learning Homework 5

Eric Larsen, A02176917

March 2024

Collaboration

For this assignment I worked through issues at the TA's office hours and with Cody Grogan. I also used Chatgpt to understand more about concepts and help fix bugs in my code as I was developing it.

Problem 1: PHATE paper

This paper goes through the motivation and implementation of PHATE and how it differs from other methods. Like we discussed in class the beginning of this paper talks about how local and global relationship data can be missing from other standard implementations of dimensionality reduction for visualization. Where t-SNE focuses on local structure, and PCA focuses on the global structures. Both of these have issues being able to retain all information. The implementation of PHATE is going through Distance calculations first, then it goes though and develops the affinities of these points to each other, yielding a diffusion probably when you raise it to t, then you take the informational distance and then yields the final mapping to represent it as two main components. From what it seems PHATE will only reduce it to two dimensions much like the first two principle components of PCA. This allows for ease in visualization allowing for a more intuitive information representation. From everything that was shown it was two dimensional, I wonder if this process could be adapted to three dimensional space, though it would come at a high cost computationally with an additional dimension in each of the matrices solved.

One of the things I still struggle to fully wrap my mind around is the performance metric used to evaluate model performance DEMaP. There isn't much mathematically listed in the methods section or in the paper itself. The most I understand is it is a metric of the distance between the ground truth and the developed representation. Going back to the implications of PHATE like the paper stated this method has the potential to accelerate understanding of relations between data points of a dataset. Where for the biological discovery filed this would speed up discovery from more labor intensive tasks to develop these relationships. Since PHATE is able to retain both the local and global relationships it is expected that we can have more confidence in the hypotheses developed using a PHATE relation map. A tool like this is of great interest to communities with high dimensionality data looking for patterns in the data, modeling of human interest similarities, or financial interests considering an increasing number of dependencies. With all the benefits of PHATE I am more interested in knowing its shortcomings and how to optimize those to increase the functionality of the method.

Problem 2: Diffusion Maps

Part A)

The implementation of diffusion maps is shown in the submitted file Problem_2.ipynb

Part B)

Part B is implemented in the code as well see the section of the code labeled DATA GENERATION

Part C)

The dataset generated for this problem is shown in the figure below.

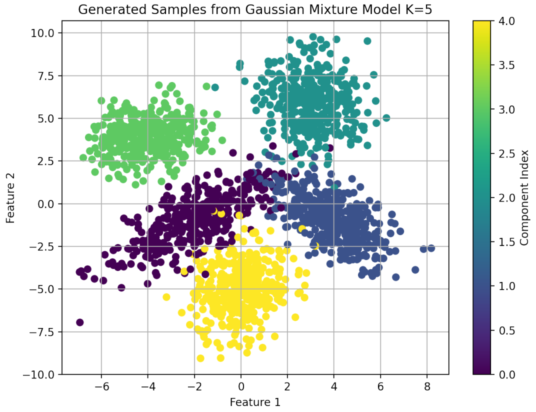

Figure 1: Generated data with K = 5

Using the method developed in the code submitted the following 5 results came from different t values each shown in its own figure

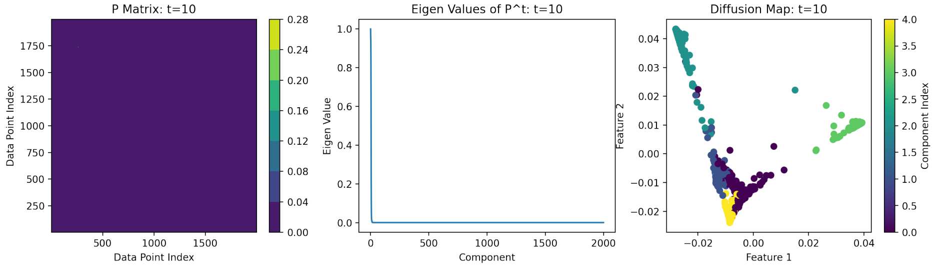

Figure 2: Diffusion Map t=10

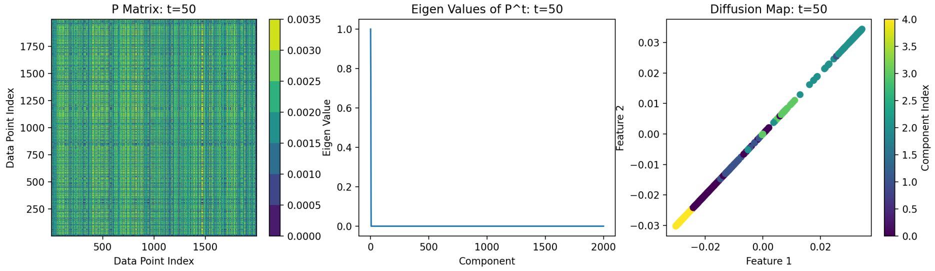

Figure 3: Diffusion Map t=50

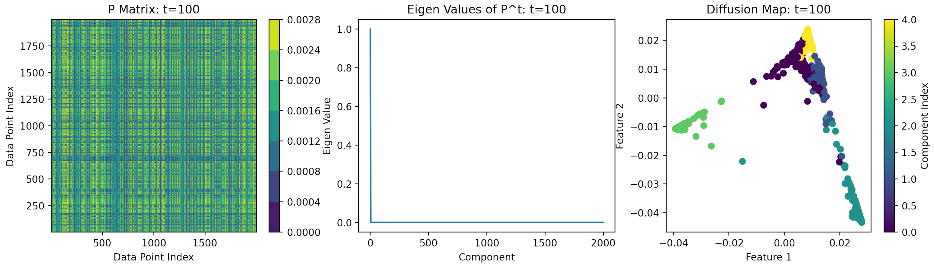

Figure 4: Diffusion Map t=100

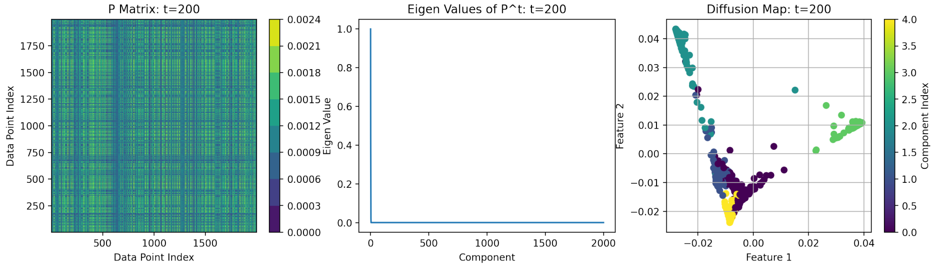

Figure 5: Diffusion Map t=200

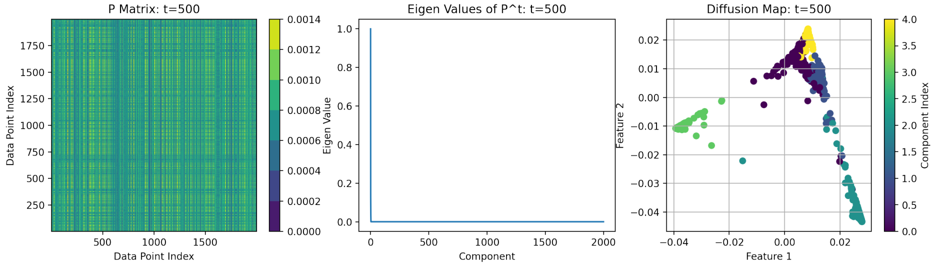

Figure 6: Diffusion Map t=500

From the dataset generated we know what we are going to need a larger sigma, where sigma is comparable to a radial distance allow for each time step. A large sigma makes the probability of different point too similar where having too small of a sigma leads more to a stationary model. From the results shown in Figures 2-6 we can see that our eigenvalues are all close to zero, with the first eigenvalue being 1 or the stationary eigenvalue. The resulting diffusion maps go to show that the model is developing a general connection well, with each group being connected to its neighbors in a correct fashion. In each of the representations it looks like the yellow class is used as the Central hub in the data with each other Gaussian distribution stemming from that cluster. Overall the general structure is the same with it generating a V or triangle shape. It was interesting to see the t = 50 case does poorly connecting all the trends, or does so in a way that it goes from one cluster to another without connecting to the previous.

Problem 3: t-SNE

Part A)

Staring with our objective function we get the following

$$KL(P||Q) = \sum_{i\neq j} p_{ij} log\left(\frac{p_{ij}}{q_i j}\right) \tag{1}$$

We know from the slides given in class our q_{ij} can be defined in the following way. In addition we know that the p_{ij} variable only depends on our X values without need of our Y variables.

$$q_{ij} = \frac{(1+||y_i-y_j||^2)^{-1}}{\sum_k \sum_{l\neq k}(1+||y_k-y_l||^2)^{-1}} = \frac{w_{ij}}{\sum_k \sum_{l\neq k} w_{kl}} \tag{2}$$

using this we can then rewrite our objective function as follows

$$KL(P||Q) = \sum_{i\neq j} p_{ij} log\left(\frac{p_{ij}}{q_i j}\right) \tag{3}$$

$$KL(P||Q) = \sum_{ij}(p_{ij} log\left(p_{ij}\right) - \sum_{ij} p_{ij}\left[log\left(w_{ij}\right) - log\left(\sum_{k\neq l} w_{kl}\right)\right] \tag{4}$$

rewritting this yields

$$KL(P||Q) = \sum_{ij}(p_{ij} log\left(p_{ij}\right) - \sum_{ij} p_{ij} log\left(w_{ij}\right) + \sum_{ij} p_{ij} log\left(\sum_{k\neq l} w_{kl}\right) \tag{5}$$

rearranging this

$$KL(P||Q) = -\sum_{ij} p_{ij} log\left(w_{ij}\right) + \sum_{ij}(p_{ij} log\left(p_{ij}\right) + \sum_{ij} p_{ij} log\left(\sum_{k\neq l} w_{kl}\right) \tag{6}$$

where we know the second term does nothing for optimizing the objective function since it has not dependence on the Y values, and instead only depends on X. Due to this we can omit it from the equation without losing the ability to optimize the objective function. The third term summation does not depend on i or j so the summation of the probability is equal to 1. This allows us to produce the following equation.

$$KL(P||Q) = -\sum_{ij} p_{ij} log\left(w_{ij}\right) + log\left(\sum_{k\neq l} w_{kl}\right) \tag{7}$$

The first term is trying to make the distances as close as possible between the old representation and the new representation. Where as the second term is controlling how close the distances can be. Where lambda commands what your distances can be, if you have a large lambda your distances need to be large so it captures the global information. Whereas with a small lambda the distances need to be small thus encoding the more local information between points. So the lambda dictates what kind of information you are going to encode.

Part B)

Taking the solution from part A we can then take the gradient in sections and use the chainrule to solve this. Taking the gradient of the first component of the objective function yields, for this we will denote the objective function as f.

$$\nabla_{y_i} f = \frac{\partial f}{\partial w_{ij}} \frac{\partial w_i j}{\partial y_i} \tag{8}$$

$$\frac{\partial f}{\partial w_{ij}} = \sum_{ij} p_{ij} \frac{1}{w_{ij}} + \frac{1}{w_{kl}} \tag{9}$$

$$\frac{\partial w_{ij}}{\partial y_i} = 2(y_i - y_j) \tag{10}$$

Since we know that k and l will only be important when $k = 1$ and $l = j$ we can then change our w_{kl} to a w_{ij} because this denotes when this condition is met. putting this all together yields the gradient of the first component of our objective function with respect to y_i is

$$\frac{\partial f}{\partial y_i} = \sum_{ij} p_{ij} \frac{2(y_i - y_j)}{w_{ij}} + \frac{2(y_i - y_j)}{w_{ij}} = \sum_{ij} p_{ij} \frac{2(y_i - y_j)}{(1 + ||y_i - y_j||^2)^{-1}} + \frac{2(y_i - y_j)}{(1 + ||y_i - y_j||^2)^{-1}} \tag{11}$$

Where the second term only matters where our k and l conditions are met with i and j points.

Problem 4: PHATE and Clustering

Part A)

The code to produce the following results is included in the assignment submission as Problem_4.ipynb. Implementing PHATE on the MNIST dataset produced the following plots.

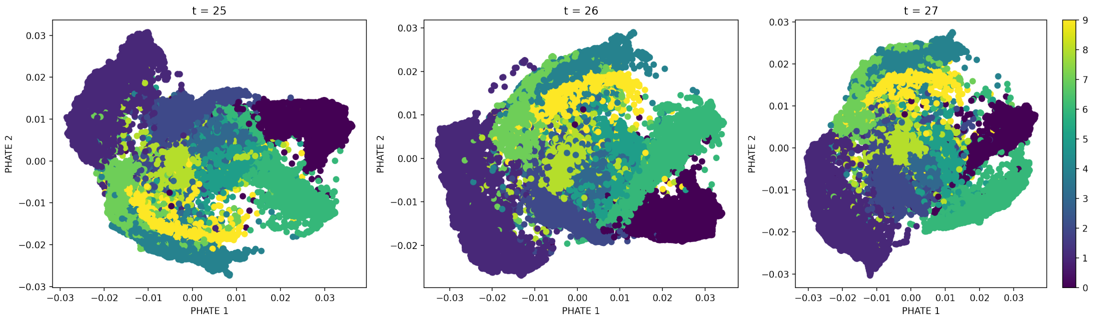

Figure 7: Results of part A using three different t values

The optimal t value found in the PHATE method was

$$t_{opt} = 25 \tag{12}$$

Though from the results of figure 7 we can see that going up and down one t value does not change the results drastically. All three of the plots look as through they are clumping the data too closely together that you can not see 10 distinct clusters. The connection between clusters doesn't make a lot of sense either where the path between values seems to instead congregate at the center and then branch out to the separate clusters. The locations of similar clusters could be attributed to similar patterns in the shape of the numbers, but for the most part these plots show no major defining features of the dataset.

Part B)

Applying KNN clustering to the dataset, subsamples were used of size 3000 with a total of 10 subsamples used. The average ARI score of these and the resulting plot are shown below.

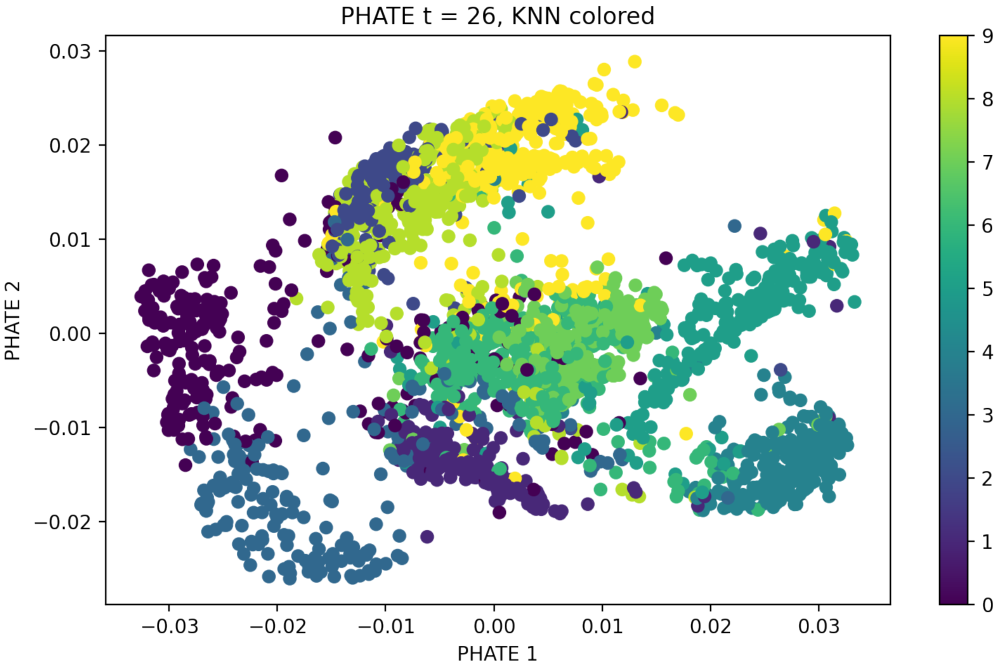

Figure 8: KNN clustering on MNIST dataset, ARI = 0.366

$$ARI_{KNN} = 0.366 \tag{13}$$

From the clustering and the ARI value we can see that the KNN method is producing better than random guessing results on the cluster labeling. I would say these results show that KNN is performing well for this task, though it is far from ideal. In the figure you can see distinct groupings form and they have good separation. This figure also shows a decent level of similarity between the true labels and the KNN labels.

Part C)

Performing another clustering method, spectral clustering, using the same subsample structure with n=10 subsamples and k = 3000 instances the following labeled plot and ARI value were found.

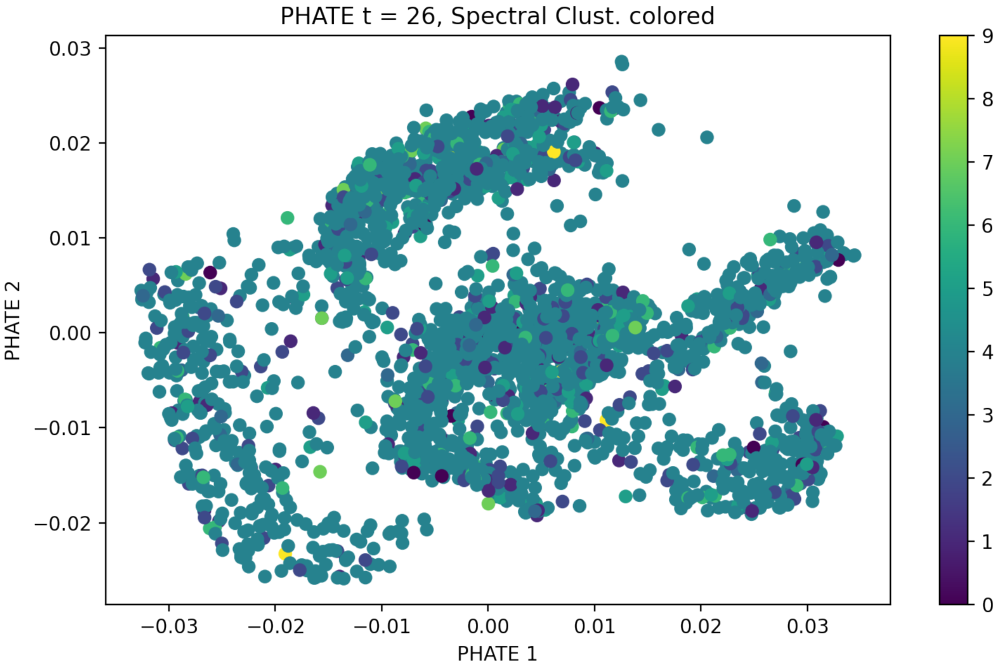

Figure 9: Spectral clustering on MNIST dataset, ARI = 0.000085

$$ARI_{Sp} = 0.000085 \tag{14}$$

From these results it is clear to see that Spectral Clustering is a poor solution for clustering MNIST data. This method performed much worse than the KNN method. Spectral clustering has problems due to method not being fully connected for this dataset. I used multiple values of gamma to set my range between points and could not generate a decent looking graph. From these results it is clear to see there needs to be an intermediate step before doing spectral clustering on the data to get the dataset into something more suitable.

Part D)

Running the PHATE method with 10 components allows us to get a new embedding to use in other methods reducing the dimensionality of the data. The optimal t value for the new PHATE method was

$$t_{opt} = 26 \tag{15}$$

Applying KNN to this new embedding of the data yielded the following plot and ARI value

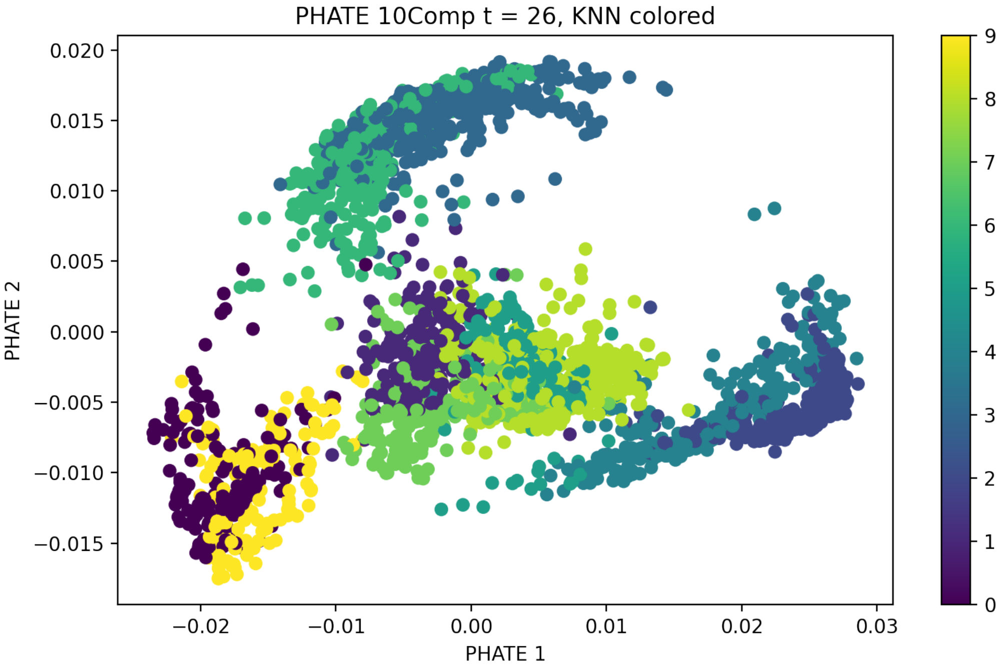

Figure 10: KNN clustering on MNIST dataset, ARI = 0.656

$$ARI_{KNN} = 0.656 \tag{16}$$

From the produced figure and the new ARI value it is clear to see this method performs better than the original KNN and spectral clustering. With this new representation we can see clearer separation between the clusters with only minor cross over. From the results of these three methods it is clear to see using the implementation given in Part D is the clear choice for applying clustering to our dataset.

Part E)

Generally it is true when applying unsupervised methods you will not have access to your expected labels. After researching this for a bit it seems to be similar to our method for finding the optimal t value. One way of doing this is called the elbow method, you vary your bandwidth parameter and see where the rate of decrease levels out when measuring your clustering score on the dataset. You want to find the optimal location to give you the best clustering score, but not push yourself into a domain of diminishing returns for increased computation cost. Another way talked about is the silhouette score, a measure of how similar a cluster is to itself and dissimilar to other clusters. Unlike the clustering score the silhouette score needs to be maximized to find the optimal bandwidth parameter.

Problem 5: Information Theory

Part A)

starting with a multi variant Gaussian the equation of this distribution is

$$p(x) = \frac{1}{\sqrt{(2\pi)^d|\Sigma|}} exp\left(\frac{-1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right) \tag{17}$$

Thus when we do the diffential entropy of this probability distribution we get the following

$$\begin{aligned} h(X) &= -E[log(p(x))] \tag{18} \\ &= -E\left[log\left(\frac{1}{\sqrt{(2\pi)^d|\Sigma|}}exp\left(\frac{-1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right)\right)\right] \tag{19} \end{aligned}$$

distributing the log yields

$$\begin{aligned} h(X) &= -E\left[log\left(\frac{1}{\sqrt{(2\pi)^d|\Sigma|}}exp\left(\frac{-1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right)\right)\right] \tag{20} \\ &= -E\left[log\left(\frac{1}{\sqrt{(2\pi)^d}}\right)log\left(\frac{1}{\sqrt{|\Sigma|}}\right)-\frac{1}{2}\left((x-\mu)^T\Sigma^{-1}(x-\mu)\right)\right] \tag{21} \\ &= -E\left[-\frac{d}{2}log\left(2\pi)\right)-\frac{1}{2}log\left(|\Sigma|\right)-\frac{1}{2}\left((x-\mu)^T\Sigma^{-1}(x-\mu)\right)\right] \tag{22} \\ &= \frac{d}{2}log\left(2\pi)\right)+\frac{1}{2}log\left(|\Sigma|\right)+\frac{1}{2}E\left[(x-\mu)^T\Sigma^{-1}(x-\mu)\right] \tag{23} \end{aligned}$$

now we have to deal with the last term where we can see using the trace operator can yield the following

$$\begin{aligned} E\left[(x-\mu)^T\Sigma^{-1}(x-\mu)\right] &= E\left[tr\left((x-\mu)^T\Sigma^{-1}(x-\mu)\right)\right] \tag{24} \\ &= E\left[tr\left(\Sigma^{-1}(x-\mu)(x-\mu)^T\right)\right] \tag{25} \\ &= tr\left(\Sigma^{-1}E\left[(x-\mu)(x-\mu)^T\right]\right) \tag{26} \\ &= tr\left(\Sigma^{-1}\Sigma\right) \tag{27} \\ &= tr\left(I_d\right) \tag{28} \\ &= d \tag{29} \end{aligned}$$

putting this back into our expression shown in equation 23 yields

$$h(X) = \frac{d}{2}log\left(2\pi)\right)+\frac{1}{2}log\left(|\Sigma|\right)+\frac{d}{2} \tag{30}$$

where this can be rewritten into the form of the equation shown in the problem of

$$h(X) = \frac{d}{2}log\left(2\pi e\right)+\frac{1}{2}log\det\left(\Sigma\right) \tag{31}$$

Part B)

Starting out with the definition of mutual information of our two distributions we can show that

$$I(X,Y) = \int P(X,Y)log\left(\frac{P(XY)}{P(X)P(Y)}\right) \tag{32}$$

expanding this equation

$$I(X,Y) = \int P(X,Y)log(P(X,Y)) - P(X,Y)log(P_x(X)P_y(Y)) \tag{33}$$

Here we can marginalize our distribution of P_x and P_y to give us another distribution we will call $\overline{P}(X,Y)$ where

$$\begin{aligned} P(X,Y) &= N(\mu,\Sigma) \tag{34} \\ \overline{P}(X,Y) = \overline{P}(Z) &= N(\mu_z,\overline{\Sigma}) \tag{35} \\ \Sigma &= \begin{bmatrix} \Sigma_x & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_y \end{bmatrix} \tag{36} \\ \overline{\Sigma} &= \begin{bmatrix} \Sigma_x & 0 \\ 0 & \Sigma_y \end{bmatrix} \tag{37} \end{aligned}$$

Rearranging equation 33 allows us to break this into two parts and see the differential entropy of the first term shown below. Where we use the result from part A for the differential entropy of the first term

$$I(X,Y) = \int P(X,Y)log(P(X,Y)) - P(X,Y)log(P_x(X)P_y(Y)) \tag{38}$$

$$I(X,Y) = -\left(-\int P(X,Y)log(P(X,Y))\right) + \int P(X,Y)log(P_x(X)P_y(Y)) \tag{39}$$

$$I(X,Y) = -\left[\frac{d}{2}log\left(2\pi e\right) + \frac{1}{2}log\det\left(\Sigma\right)\right] + \int P(X,Y)log(P_x(X)P_y(Y)) \tag{40}$$

Now we need to focus on the second term of the equation to get it's own differential entropy, using our new distribution leads us to solving the following where the distribution Z is equal to our marginal distribution of X and Y.

$$P(Z) = \int P(Z)log(\overline{P}(Z) \tag{41}$$

$$h(Z) = -E[log(P(Z)] \tag{42}$$

where part A we know this will end up with form

$$h(Z) = \frac{d}{2}log(2\pi) + \frac{1}{2}log(det(\overline{\Sigma})) + \frac{1}{2}E\left[(z-\mu_z)^T\overline{\Sigma}^{-1}(z-\mu_z)\right] \tag{43}$$

focusing on the last term we can use the same trick with the trace as we did before yielding the following

$$E\left[(z-\mu_z)^T\overline{\Sigma}^{-1}(z-\mu_z)\right] = E\left[(z-\mu_z)^T\overline{\Sigma}^{-1}(z-\mu_z)\right] \tag{44}$$

$$= E\left[tr\left(\overline{\Sigma}^{-1}(z-\mu_z)(z-\mu_z)^T\right)\right] \tag{45}$$

$$= tr\left(\overline{\Sigma}^{-1}E\left[(z-\mu_z)(z-\mu_z)^T\right]\right) \tag{46}$$

$$\tag{47}$$

Here is is important to recognize that z is the total distribution, so the expectation of the this term will give us the covariance matrix for the total domain, our original Σ matrix. Using this we can then see this continues to be.

$$= tr\left(\overline{\Sigma}^{-1}\Sigma\right) \tag{48}$$

$$= tr\left(\begin{bmatrix}\Sigma_x^{-1} & 0 \\ 0 & \Sigma_y^{-1}\end{bmatrix}\begin{bmatrix}\Sigma_x & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_y\end{bmatrix}\right) \tag{49}$$

$$= tr\left(\begin{bmatrix}I_d & 0 \\ 0 & I_d\end{bmatrix}\right) \tag{50}$$

$$= d \tag{51}$$

putting this result back into equation 43 and then sub that back into equation 40 we get the following

$$I(X,Y) = -\left[\frac{d}{2}log\left(2\pi e\right) + \frac{1}{2}log\det\left(\Sigma\right)\right] + \left[\frac{d}{2}log(2\pi) + \frac{1}{2}log(det(\overline{\Sigma})) + \frac{d}{2}\right] \tag{52}$$

$$I(X,Y) = -\left[\frac{d}{2}log\left(2\pi e\right) + \frac{1}{2}log\det\left(\Sigma\right)\right] + \left[\frac{d}{2}log(2\pi e) + \frac{1}{2}log(det(\overline{\Sigma}))\right] \tag{53}$$

$$I(X,Y) = -\frac{1}{2}log\det\left(\Sigma\right) + \frac{1}{2}log(det(\overline{\Sigma})) \tag{54}$$

$$I(X,Y) = \frac{1}{2}log\left(\frac{det(\overline{\Sigma})}{det(\Sigma)}\right) \tag{55}$$

Therefore we can see that our mutual information between X and Y is

$$I(X,Y) = \frac{1}{2}log\left(\frac{det(\overline{\Sigma})}{det(\Sigma)}\right) \tag{56}$$

Problem 6: Outlier Detection with Kernel Density Estimation

Part A)

This method was done and is shown in the code submitted under part A. The optimal bandwidth found for this dataset from the code submitted was

$$h_{opt} = 0.0793 \tag{57}$$

Part B)

The generated density function f using the optimal bandwidth found from -2 to 4 yields the following plot. Note that the data distribution as a bar plot is overlaid on the data structure to give an understanding of how the distribution should look.

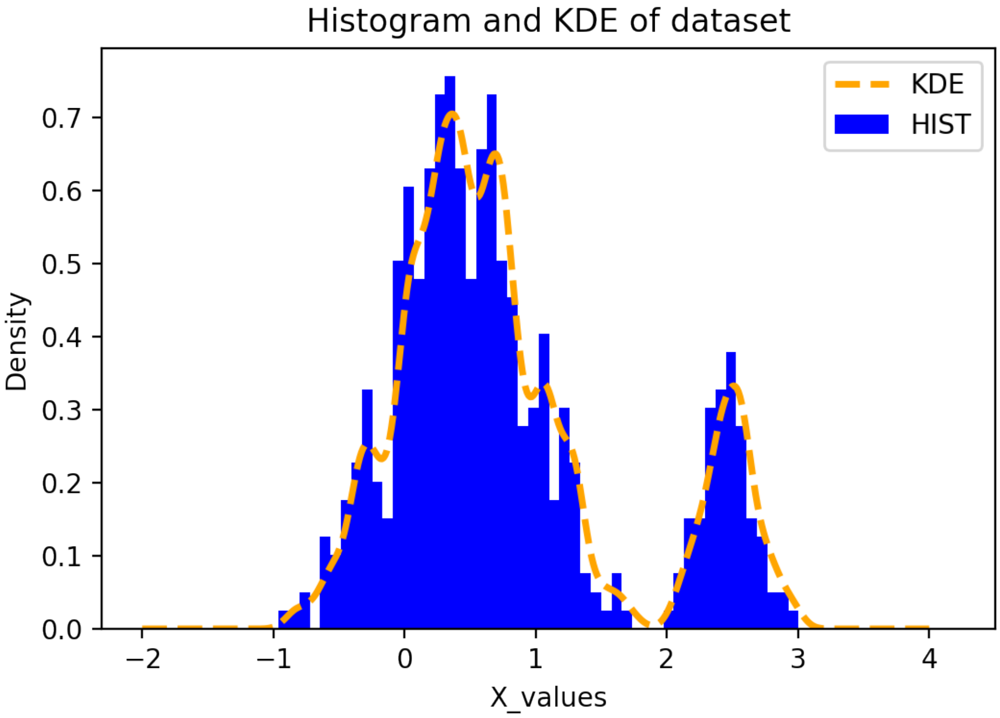

Figure 11: estimated density function f from KDE

From this distribution we can see that the expected density function looks like a multi modal Gaussian distribution with peaks around 0.5,0.75, 2.5 etc.

Part C)

The interpretation of this outlier score is measuring the ratio of the probability density of the test point compared to the average density of the training set. Therefore, if the outlier score is low the probability of the point compared to the average density of the training points is low and the point is most likely an outlier. Where as if the value is high we know the probability of the point in the distribution compared to the average is high, so the point most likely belongs to the distribution that created the dataset. The main weakness of this outlier method is its dependence on our bandwidth parameter selected for the KDE method. If we have too large of an h we over smooth the distribution and may give incorrect results for this outlier score.

Part D)

The method to calculate the outlier score 1 for these two test points is given in the code file submitted with this assignment. The results yield the following

$$\text{Outlier1}(x_1) = 0.1714232 \tag{58}$$

$$\text{Outlier1}(x_2) = 1.581e - 26 \tag{59}$$

$$\tag{60}$$

From these results we can assume that test point 1 is not an outlier with a decently high ratio between the probability and average probability. Where as test point 2 is most likely an outlier for this dataset with the ratio of probability being basically zero.

Part E)

The conceptual interpretation of this score is the distance to the kth nearest neighbor compared to the average distance to all other K neighbors Kth neighbor. At the basics it is checking to see how far the distance to the kth neighbor compares as a ratio to the mean distance of a collection of other Kth neighbors. If the value is high we know the value is an outlier because the distance ratio is large, it is unlike the other values. Where as if the outlier score is small then we know that it is not an outlier because the distance to the point is similar to the average distances to the neighboring points. Advantages of this method compared to the former is its insensitivity to bandwidth, we are checking relative distances compared to the neighborhood instead of the actual distance. So as we change the bandwidth this value should not change much if at all. This also focuses on the local area of the point instead of the point in a global sense.

Part F)

Using the second outlier score for different K values yields the following result from the code.

Table 1: Outlier2 Scores for xtest1 and xtest2 points

Dataset	**k**	**Outlier Score**
xtest1	100	1.4188
xtest2	100	4.4209
xtest1	150	1.7129
xtest2	150	4.3071
xtest1	200	1.9555
xtest2	200	3.9948

From these results it is clear to see that second point is an outlier where as the first point looks similar to its neighboring points so it is not likely an outlier.

Part G)

The code is included in the submission as Problem_6.ipynb

References