DREAM Fellowship Final Report

Quentin Chu

Summer-Fall 2023

Structure Preserving Metric Learning

1. Introduction

Metric learning is a set of machine learning problems that aim to construct task-specific distance metrics from supervised data. The most common metric learning methods seek to learn a global mapping of the data set and apply the same distance function on the transformed data. In doing so, a mapping that is effective in achieving a specified task may destroy other underlying information about the data that is not needed to the immediate task, but may be relevant for subsequent investigation. Structure Preserving Metric Learning (SPML) aims to learn the data mapping that provides the most accurate measure of similarity, while retaining the structure of each similar group to allow subsequent exploration of subgroup features.

2. Related Work

There is a significant amount of existing literature on metric learning, much of which is chronicled in the survey papers by Kulis (2012) and Bellet et al. (2014). We have specifically reviewed MMC (Xing et al, 2002), NCA(Goldberger et al., 2005), t-SNE (van der Maaten and Hinton, 2008) and LMNN (Weinberger and Saul, 2009), and are using a modified implementation of MMC in our SPML algorithm.

3. Structure Preserving Metric Learning

The SPML algorithm is based on a modified version of the existing metric learning MMC algorithm, with an additional term representing the structural integrity of the subgroups in

the objective function. Thus, the objective function and constraints are as follows:

$$\begin{aligned}\min_{A} \quad & \sum_{(x_i,x_j)\in S} ||x_i - x_j||_A^2 - \sum_{x_c \in C} \texttt{struct}(A^{1/2}\boldsymbol{x_c}, k) \\ \text{s.t.} \quad & \sum_{(x_i,x_j)\in D} ||x_i - x_j||_A \geq 1, \\ & A \succeq 0\end{aligned}$$

where `struct` provides the structural integrity score of the transformed data in class c given k subclusters.

While there are many existing methods that can identify clusters in unlabeled data, including Gaussian Mixture Model and Dirichlet Process Mixture Model, the resulting loss function would not be differentiable if these methods are called. Therefore, we propose Distance Concentration Score as a definition of `struct` to provide an approximate representation of structural integrity when the number of subclusters in each class is known.

Distance Concentration Score

This definition of `struct` takes the transformed data and the known number of subclusters in each class, computes a distance matrix for all data points, and returns a structural integrity score based on the how close the average pairwise distance is to the optimal average pairwise distance. The optimal average pairwise distance assumes that the data points are divided evenly among the given number of subclusters, which are separated evenly among themselves.

Let $A^{1/2}\boldsymbol{x_c}$ be a $d \times n$ matrix representing the transformed data in class c, k be the number of subclusters in the class, and D be the distance matrix among all data points in the class. The `struct` score is defined as follows:

$$\texttt{struct}(A^{1/2}\boldsymbol{x_c}, k) = \exp\left(-\frac{(NormDist - OptDist)^2}{\sigma^2}\right)$$

$$\begin{aligned}\text{where} \quad NormDist &= \frac{\sum_{i,j}^{n} \sqrt{\frac{1}{\mu} \cdot D_{i,j}}}{n^2}\ , \\ \mu &= \frac{\sum_{i,j}^{n} D_{i,j}}{n^2}\ , \\ D_{i,j} &= ||x_i - x_j||_A = \sqrt{(x_i - x_j)^\intercal A (x_i - x_j)}\ , \\ OptDist &= \sqrt{\frac{k-1}{k}}\ , \\ \sigma &= \frac{\sqrt{\frac{k}{k-1}} - \sqrt{\frac{n}{n-1}}}{3}\end{aligned}$$

This `struct` definition is based on the observation that if all clusters are divided and separated evenly, then there is an optimal average pairwise distance such that distances among

data points within the same clusters are minimized and distances among data points from different clusters are maximized. To calculate the normalized average pairwise distance ($NormDist$), we divide all entries of the distance matrix by the mean distance, take the square root of each entry before taking the summation, and divide the total by n^2, which is the number of entries in the distance matrix. Taking the square root of each distance entry ensures that $NormDist$ does not always equal to 1.

The optimal average pairwise distance is $(k-1)/k$. We derive $OptDist$ by taking the square root of the optimal average distance between data points from different clusters, multiplying it by the number of non-zero entries in an optimal distance matrix, and dividing it by n^2. Thus we have:

$$\begin{aligned} OptDist &= \frac{\left(\frac{n}{k}\right)^2 \cdot (k^2-k) \cdot \sqrt{\frac{k}{k-1}}}{n^2} \\ &= \sqrt{\frac{k-1}{k}} \end{aligned}$$

We then apply a Gaussian function centered around the optimal distance to compute the final score. Since we know that a perfectly scattered set of data points (that should result in a low `struct` score) has an average normalized pairwise distance of $n/(n-1)$, we set σ such that $\sqrt{\frac{k}{k-1}} - \sqrt{\frac{n}{n-1}}$ represents 3 standard deviations from the optimal score.

Loss Function Using Modified MMC

Instead of a hard constraint on the distance between dissimilar points, we implement a modified version of MMC by imposing a soft constraint in the following loss function:

$$\mathcal{L} = \lambda_1 \sum_{(x_i,x_j)\in S} ||x_i - x_j||_A^2 - \lambda_2 \sum_{(x_i,x_j)\in D} ||x_i - x_j||_A - \lambda_3 \sum_{x_c \in C} \texttt{struct}(A^{1/2}\boldsymbol{x_c}, k)$$

We normalize the sums of distances by setting $\lambda_1 = \frac{1}{|S|}$ and $\lambda_2 = \frac{1}{|D|}$, and set λ_3 to be proportional to the squared number of dimensions (d^2), the number of classes (k) and subclusters per class (s) to scale with the data complexity. Thus we have the derivative with respect to A as follows:

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial A} = \frac{1}{|S|} \sum_{(x_i,x_j)\in S} (x_i - x_j)(x_i - x_j)^\intercal - \frac{1}{|D|} \sum_{(x_i,x_j)\in D} \frac{(x_i - x_j)(x_i - x_j)^\intercal}{2\sqrt{(x_i - x_j)^\intercal A (x_i - x_j)}} \\ - d^2 ks \cdot \frac{\partial\, \texttt{struct}}{\partial A} \end{aligned}$$

Differentiating `struct`

Since $NormDist$ is the only part of `struct` that is dependent on A, we see that:

$$\frac{\partial \texttt{struct}}{\partial A} = \exp\left(-\frac{(NormDist - OptDist)^2}{\sigma^2}\right) \cdot \frac{-2(NormDist - OptDist)}{\sigma^2} \cdot \frac{\partial}{\partial A}\left(\frac{\sum_{i,j}^{n} \sqrt{\frac{1}{\mu} \cdot D_{i,j}}}{n^2}\right)$$

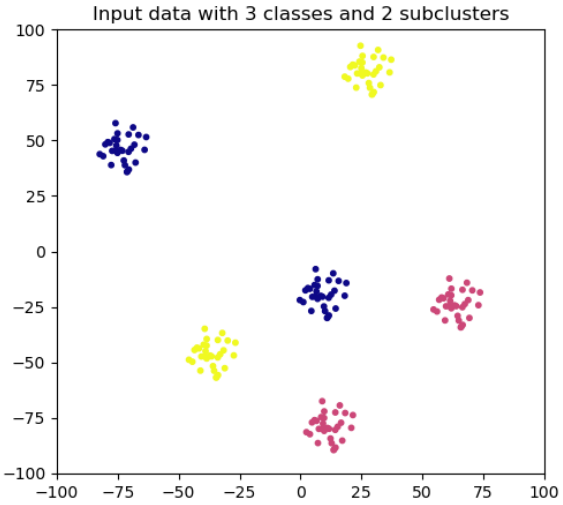
where

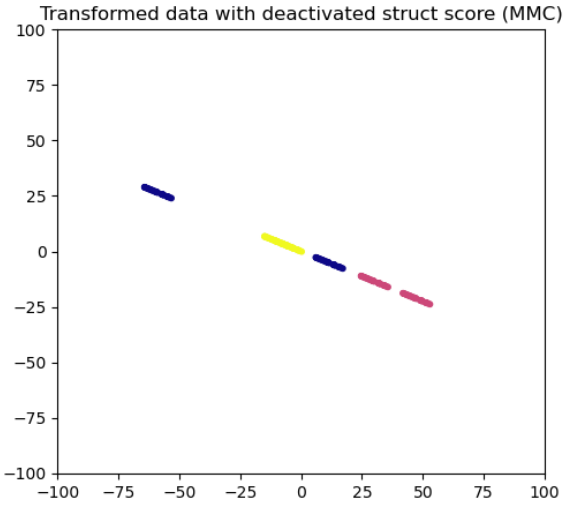
$$\begin{aligned}\frac{\partial}{\partial A}\left(\frac{\sum_{i,j}^{n} \sqrt{\frac{1}{\mu} \cdot D_{i,j}}}{n^2}\right) &= \frac{\partial}{\partial A}\left(\frac{\sum_{i,j}^{n} \sqrt{D_{i,j}}}{\sqrt{\sum_{i,j}^{n} D_{i,j}}}\right) \\ &= \left(\sum_{i,j}^{n} \frac{(x_i - x_j)(x_i - x_j)^\intercal}{4(D_{i,j})^{3/2}} \cdot \sqrt{\sum_{i,j}^{n} D_{i,j}} - \sum_{i,j}^{n} \sqrt{D_{i,j}} \cdot \frac{\sum_{i,j}^{n} \frac{(x_i - x_j)(x_i - x_j)^\intercal}{2D_{i,j}}}{2\sqrt{\sum_{i,j}^{n} D_{i,j}}}\right) \cdot \frac{1}{\sum_{i,j}^{n} D_{i,j}}\end{aligned}$$

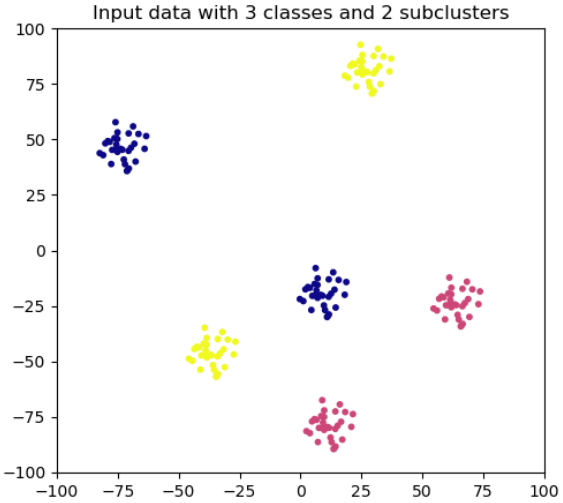
With the derivative of the SPML loss function with respect to A, we perform stochastic gradient descent on A, while projecting A onto the set of positive semi-definite matrices in each iteration to enforce the hard constraint of $A \succeq 0$.

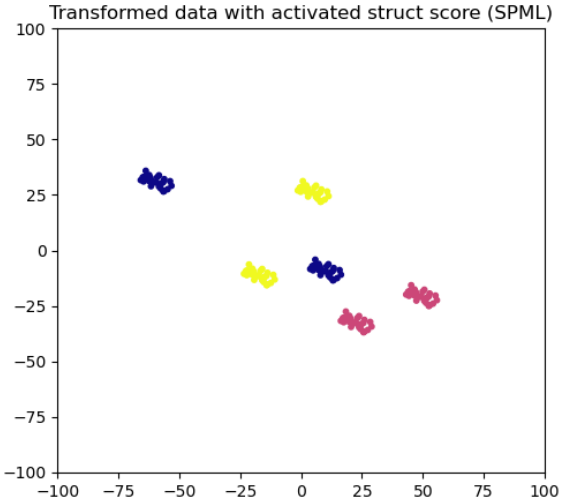
4. Experiments

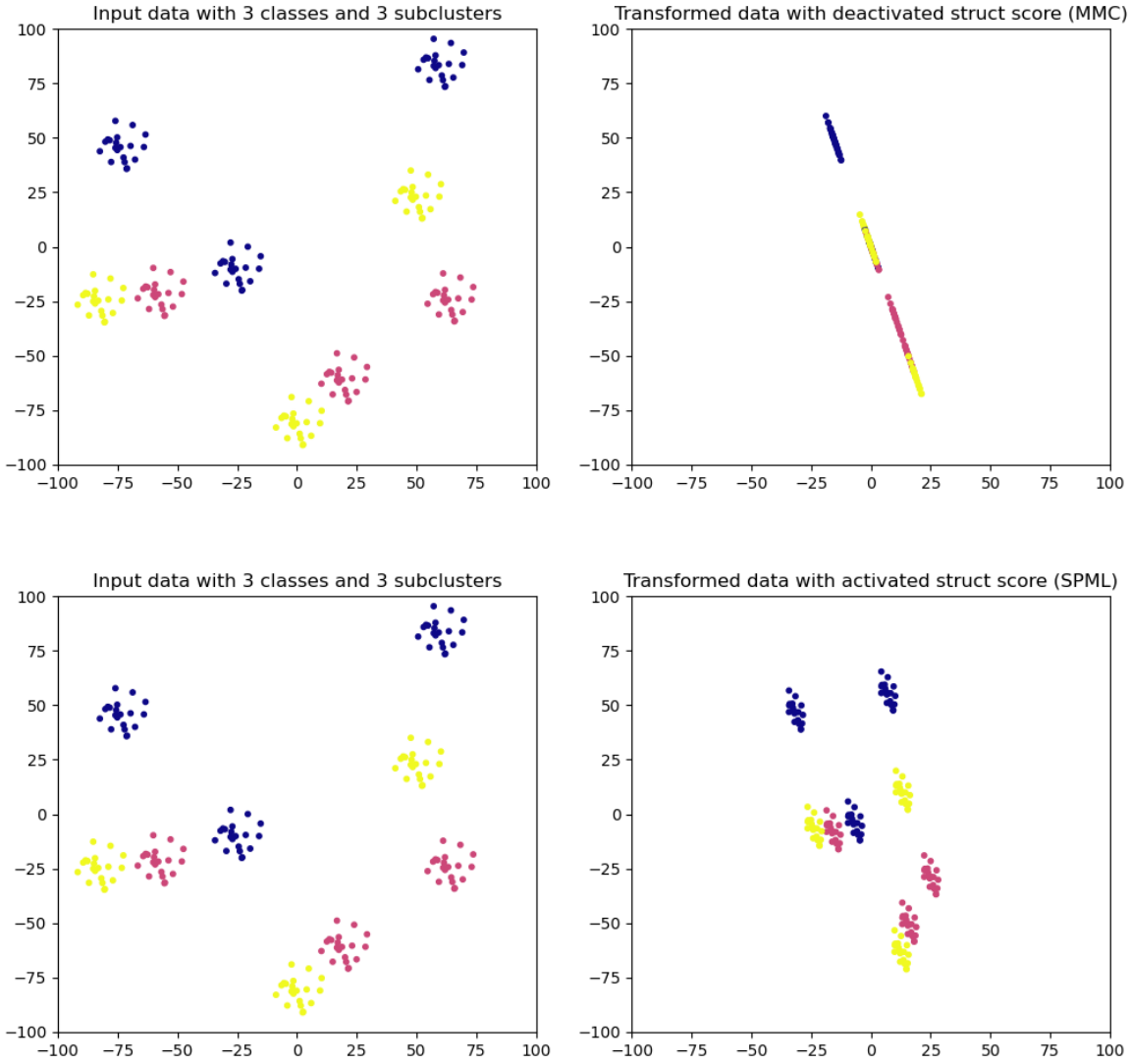
We created two sets of input data in 2 dimensions to compare the visual effects of data transformation via MMC versus SPML. The first set of data consists of 3 labeled classes and 2 subclusters in each class, while the second set has 3 classes and 3 subclusters per class. Using the same SPML algorithm for all experiments, we simulate MMC by setting λ_3 to a low value and deactivating the `struct` loss function. We see that MMC collapses each labeled clusters into one-dimensional clusters on a line, while SPML brings the transformed data closer together but retains the separation among subclusters.


Input data with 3 classes and 2 subclusters


Transformed data with deactivated struct score (MMC)


Input data with 3 classes and 2 subclusters


Transformed data with activated struct score (SPML)



5. Conclusion

The results show that it is possible to append a differentiable function that retains the underlying structure (defined as existence of a specified number of subclusters) in the form of the proposed `struct` function. However, the underlying MMC classifier fails to separate the different classes distinctly, such that the error rate for any test data in the domain could still be significant. Future extensions could include using other metric learning methods (such as LMNN) as the basis of SPML.