DATA CLUSTERING WITH
COMMUTE TIME DISTANCE

Abhirupa Sen
Supervisor: Dr. Guangliang Chen
Contents

- Random walk with respect to a graph
- Transitional Probabilities
- Commute Time Distance
- Clustering of toy data using commute time distance
- Clustering of handwritten digits 0 and 1 using commute time distance.
- Significance of tuning
- Illustration of the improvement on tuning
Graph Laplacian

- Graph Laplacian $L$ is a measure of pairwise connectivity of the vertices.
- For two vertices $i$ and $j$, $L_{ij}$ has non zero value if $i$ and $j$ are neighbors.
- The diagonals of $L$ specify the total degree of a vertex/point.
- By degree we mean how connected a point is.
- An vertex with zero degree is an isolated point on the graph with no neighbors.
- A point with zero degree can be thought of as an outlier.
Random Walk

- A graph with n vertices is like a path with n intersections
- At an intersection a walker can choose a path randomly.
- Each path has a probability of being chosen. This is called **transitional probability**.
- Transitional probability of an edge is directly proportional to the weight of the edge.
  => Edges with higher weight have higher probability of being chosen
  => Edges have higher weight when vertices are closer/similar to each other.
- A random walker tends to jump between similar vertices.
- Similar vertices means vertices belonging to a cluster.
Transitional Probabilities

- Transitional Probabilities
  
  \[ p_{ij} = P(X_{t+1} = v_j \mid X_t = v_i) = \frac{\text{weight of connection between } v_i \text{ and vertex } v_j}{\sum \text{weight of all the edges to } v_i} = \frac{w_{ij}}{d_i} = D^{-1}W_{ij} \]

- \( D^{-1}W \) is the transitional probability matrix \( P \).
- \( D \) is diagonal with \( D_{ii} \) specifying the degree of vertex \( i \) = total weight of all the connections.
- \( D^{-1} \) is diagonal too with \( D^{-1}_{ii} = 1/D_{ii} \).
- \( L_{RW} = D^{-1}L = D^{-1}(D - W) = I - D^{-1}W = I - P \)
- Higher the weight higher is the transitional probability.
- Weights are higher for edges between similar points.
  
  ⇒ **Within cluster transitional probabilities are higher.**
  ⇒ **Between cluster transitional probabilities are lower.**
Commute Time Distance

- Commute time distance is the expected time required to travel back and forth between vertices $v_i$ and $v_j$.
- More connected/similar $v_i$ and $v_j$ are lesser is this time.
- Points within a cluster must have smaller commute time distance.
- Points from two different clusters must have larger commute time distance.
- Commute time distance $\propto \frac{1}{\text{Transitional Probabilities}}$
- $P = 1 - L_{RW}$
- Commute distance is calculated using pseudo inverse of $L_{RW}$. 
Application on toy data: Three circles

3 circles clustering

kmeans clustering

MDS plot using Commute Distance Matrix

Kmeans on the MDS mapping

weights between points = $e^{-\frac{||x_i-x_j||^2}{2\sigma^2}}$

$\sigma = 0.08$
Application on data two Gaussians

- Comparison of kmeans and Commute distance clustering on a data with two

Scatter plot of dataset with 2 Gaussian clusters.

Spectral clustering using graph Laplacian.
Two Gaussians clustering using Commute Distance

MDS mapping in 2 dimensions.
The blue points are the noise data.

Clustering of the data with the outlier
Clustering of handwritten digits 0 and 1

- The data set consists of 2115 data points.
- Each of the data is an image of size 28x28.
- The images are vectorized into vectors of 784 dimensions.
- Not all these dimensions are useful.
- Using PCA the dimension was reduced to 10.
- The reduced data space is 2115x10 dimensional.
Two dimensional representation of the data
3D MDS mapping: Clustering using one $\sigma$

Digits 0 and 1 still overlap
Self Tuning

- The data points for digit 1 are tightly bound
- The data cloud of zeros is comparatively loose.
- Two $\sigma$'s are used in pairwise weight computation.

\[
\begin{align*}
\text{weights between clusters} &= e^{-\frac{||x_i-x_j||^2}{2\sigma_i\sigma_j}} \\
\text{weights within clusters} &= e^{-\frac{||x_i-x_j||^2}{2\sigma_i^2}}
\end{align*}
\]
Clustering using two $\sigma$ s for the two class

The clusters of the two digits are well separated
Three dimensional MDS mapping of the clusters.

The red cluster are the 1s and the blue cluster are the 0s.

The two outliers for digit 1.

The outlier for digit 0.
Future work

- Further improvement by point wise tuning.
- Plan to use this method to cluster the MNIST dataset of digits
- Pairwise as well as ten clusters
- The method is computationally expensive.
- Using it for image segmentation