Spectral Clustering of Time Series Data

Yingbo Song and Kapil Thadani

Advanced Machine Learning

Fall 2006

Columbia University

Overview

- Project Goals
  - Clustering time series data that contain smoothly varying HMM parameters
- Background on HMM & EM
- Current approaches to HMM clustering
- Background on spectral clustering
- The spectral clustering approach to HMM clustering
- Data used
- Current test results
- Conclusions

Project Goals

- Unsupervised learning is a well studied problem
- Typically applied to data in vectorized form
- What about variable length data? Such as time series

- Our goal:
  - Investigate time-series clustering using spectral clustering with probability product kernels
  - Compare with existing methods of time-series clustering

Background

- HMM clustering using EM
  - Train a mixture of *k* HMMs on the data
  - Calculate posteriors from likelihoods of each sequence with each HMM in E-step
  - Update the parameters of each HMM with the posteriors in M-step
  - Each sequence influences the parameters of each HMM by its posterior probability under that HMM

$$\bar{\pi}_i^{(m)} = \frac{\sum_{l=1}^{L} z_{lm} \gamma_1^{(lm)}(i)}{\sum_{l=1}^{L} z_{lm}}$$

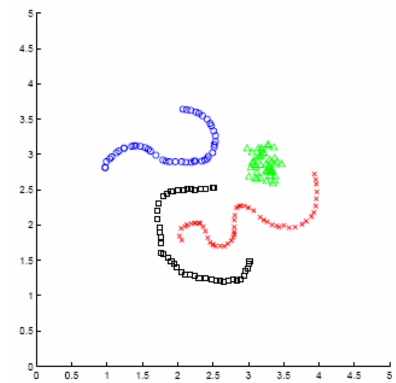
$$\bar{a}_{ij}^{(m)} = \frac{\sum_{l=1}^{L} z_{lm} \sum_{t=1}^{T-1} \xi_t^{(lm)}(i,j)}{\sum_{l=1}^{L} z_{lm} \sum_{t=1}^{T-1} \gamma_t^{(lm)}(i)}$$

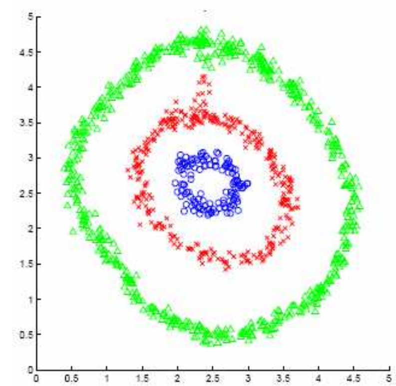
$$\bar{\mu}_j = \frac{\sum_{l=1}^{L} z_{lm} \sum_{t=1}^{T} \gamma_t^{(lm)}(i) \cdot O_t}{\sum_{l=1}^{L} z_{lm} \sum_{t=1}^{T} \gamma_t^{(lm)}(i)}$$

$$\bar{\Sigma}_j = \frac{\sum_{l=1}^{L} z_{lm} \sum_{t=1}^{T} \gamma_t^{(lm)}(i) \cdot (O_t - \mu_j)(O_t - \mu_j)'}{\sum_{l=1}^{L} z_{lm} \sum_{t=1}^{T} \gamma_t^{(lm)}(i)}$$

Background

- Spectral Clustering
  - Calculate the Gram matrix of dataset
  - Calculate the normalized Laplacian matrix
  - Find the eigenvectors of the Laplacian matrix
  - Run clustering algorithm on the components of the eigenvectors





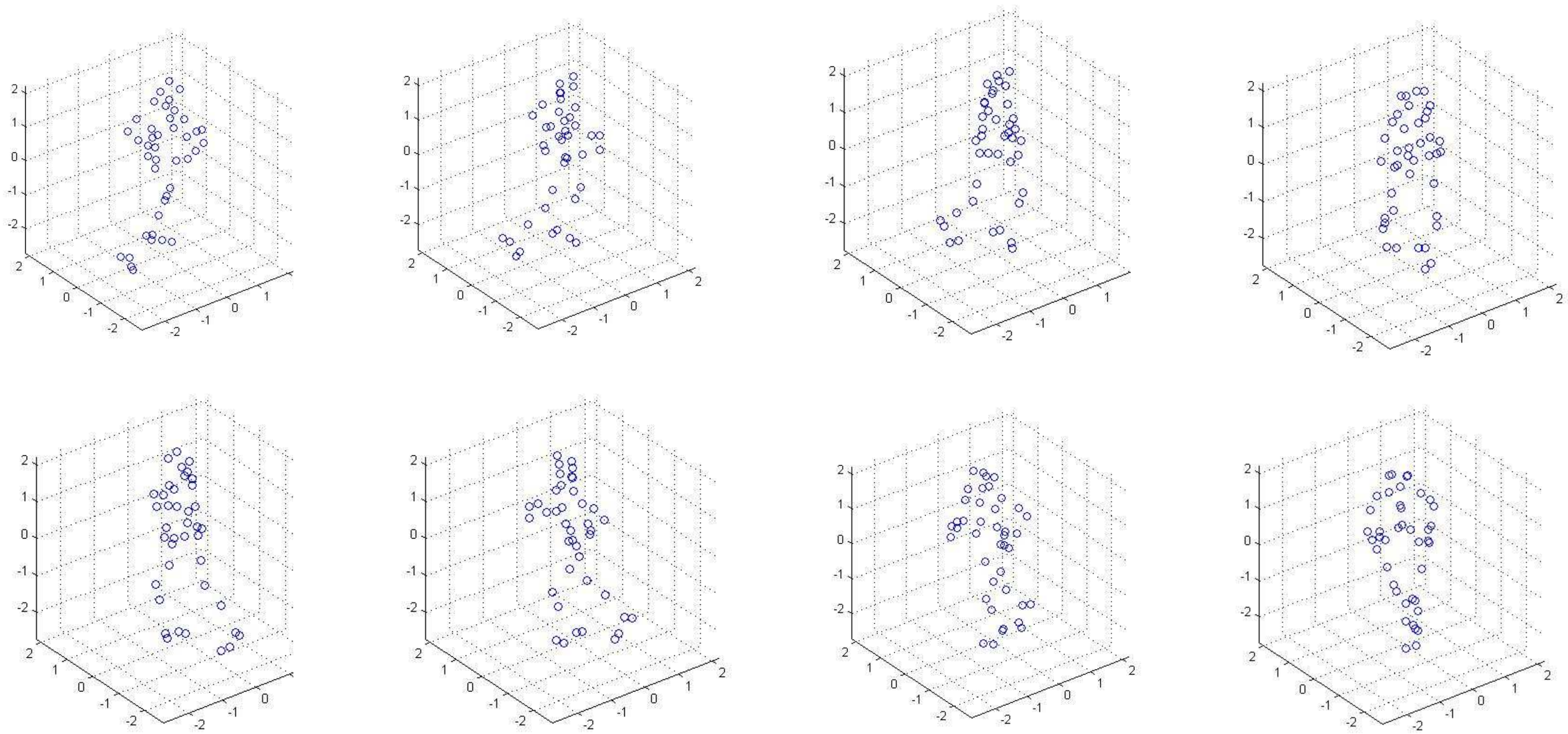
- Probability Product Kernels
  - Calculate the probability distribution of the HMM over a dataset
  - The kernel affinity is the similarity between two distributions
  - Bhattacharya affinity

Spectral Clustering of HMMs

- Train single HMMs over each data sequence
- Find probability of each data sequence over each HMM
- The elements of the gram matrix are the probability product kernel affinities
- Find eigenvectors of the normalized gram matrix
- Run K-means on the components of the eigenvectors

Data

- Rotated MOCAP Data – 360 Degrees



Results

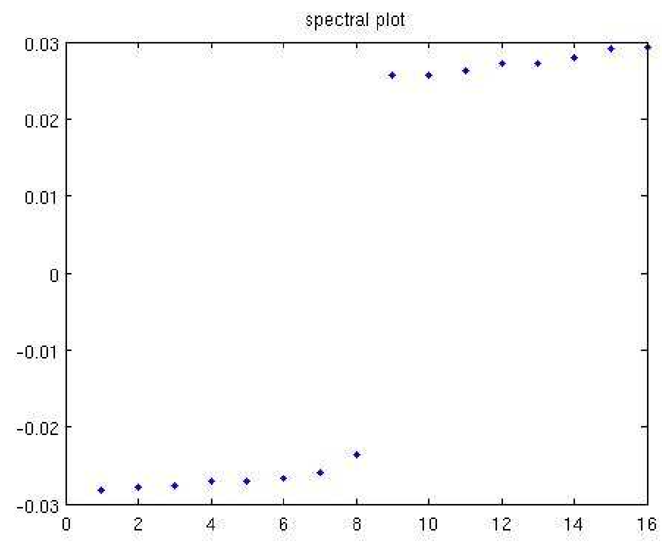
- Testing results over MOCAP data – Accuracy rates
- Sub-sampled at 5° degree step sizes

	EM-HMMs	Spectral Clustering
Non-rotated MOCAP	100%	100%
Rotated 0 - 45°	100%	100%
Rotated 0 - 90°	~50%	100%
Rotated 0 -120°	~50%	100%
Rotated 0 -150°	~50%	50%
Rotated 0 -180°	~66%	~50%
Rotated 0 -360°	~50%	~50%

- In our testing, spectral clustering was roughly twice as fast as EM-HMM
- Accuracy results were influenced by what we believe to be an Intersecting manifold in HMM parameter space

Advantages

- Speed
  - Like EM-HMM except with only one iteration
  - Can control the number of samples used to find the distribution, reduce it to boost speed
- Accuracy
  - On slowly varying data with non-gaussian HMM parameter distributions, the spectral clustering approach outperforms EM-HMM
  - EM-HMM can get stuck on local minima
- Eigengap
  - Can estimate the number of clusters by looking at the eigengap!
- Local Max
  - EM is susceptible to low max, spectral clustering is not



Conclusions

- Spectral clustering works well with slowing varying data. Can handle cases where the HMM parameters define some bendy non-Gaussian manifold.
- Results for time series clustering using spectral clustering is analogous to the results found in clustering non-time series data.

- Currently working on
  - Testing on more real world data (MOCAP, GAIT)
  - Testing on multi-class data
  - Exploring better ways to find K based on the eigengap
  - Exploring better ways to improve speed