SUBSPACE CLUSTERING

Ufuk Soylu

School of Electrical and Computer Engineering, University of Illinios Urbana-Champaign

ABSTRACT

Recently, there has been an explosion in the availability of data from multiple sources and modalities. Even though the data are generally high dimensional, their intrinsic dimension is much smaller than its original dimensionality. Therefore, this motives the development of many techniques to represent high dimensional data with lower dimensions. Conventional techniques assume that data is from single subspace. However, in practice data is from multiple subspaces. Therefore, there is a need to cluster data into multiple subspaces. This problem is known as subspace clustering which is the main concern of the project. The project is based on the work of Rene Vidal [1]. Some face clustering algorithms, which are introduced in [1], are tested for data-set generated by the author. Data-set consists of profile images of seven persons instead of frontal images which is the way that [1] tested the algorithms. The author concludes that some face clustering algorithms produce very stable and accurate results when working with profile images.

Index Terms— clustering algorithms;data models,linear subspace;affine subspace;motion segmentation;face clustering problem

1. INTRODUCTION

The main goal of the project is to explain [1], to try and reproduce results from [1], to point out the difficulties. [1] discusses about subspace clustering problem. The main motivation for subspace clustering is to reduce the dimensionality of the data.

Recently, there has been an explosion in the availability of data from multiple sources and modalities. It leads to many advances in data acquisition,compression, transmission, processing massive amount of high dimensional data. Main advances depends on the fact that even though data is high dimensional, the intrinsic dimension is much smaller which motives the development of many techniques to represent high dimensional data with lower dimensions. Conventional techniques assume that data is from one single subspace. Such techniques are used in many areas such as pattern recognition, data compression, image processing, bioinformatics etc. In practice, data points can come from multiple subspaces without knowing which point belongs to which subspace. Therefore, there is a need to cluster the data into multiple subspaces which is known as subspace clustering problem.

[1] is a survey of techniques that are related to subspace clustering problem. It presents the methods under four groups: algebraic methods, iterative methods, statistical methods and spectral clustering-based methods. Then, it evaluates the success of the methods from all classes by considering two applications in computer vision: motion segmentation from feature point trajectories and face clustering under varying illumination.

In this project, based on [1], face clustering algorithms are tested with side views(profile) of faces. The author collected 107 profile images from 7 different persons. Then, face clustering algorithms are tested on this data-set. Contribution of this article is to test face clustering algorithms with profile images. This article's outline is as follows:

- Problem Definition of Subspace Clustering.
- Subspace Clustering Algorithms
- Applications: Motion Segmentation and Face Clustering
- Results: Face Clustering for Profile Images

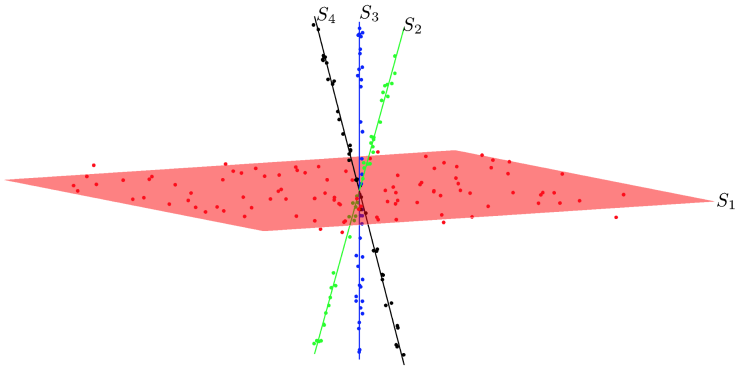

Fig. 1. An Example of Subspace Clustering

2. THE SUBSPACE CLUSTERING PROBLEM

The problem is to model a collection of data points from union of unknown number of subspaces. In Fig 1, there is

a visualization of such problem whose data points are from union of four subspaces named as S_1, S_2, S_3, S_4. The goals of the subspace clustering algorithms are to find the number of subspaces, their dimensions and their basis and to find data segmentation.

In order to express it more mathematically, let $\{\mathbf{x}_j \in \mathbb{R}^D\}_{j=1}^N$ be a given set of data points from unknown data modality and $\{S_i\}_{i=1}^n$ be linear of affine subspaces, $d_i = dim(S_i)$ and $0 < d_i < D$ be the dimensions of each subspaces. Then subspaces can be described as follows:

$$S_i = \{\mathbf{x} \in \mathbb{R}^D : \mathbf{x} = \mu_i + U_i y\}, \quad i = 1, \ldots, n$$

where $\mu_i \in \mathbb{R}^D$ is an arbitrary offset point. If $\mu_i = 0$, it means that subspace passes through the origin so it is a linear subspace, otherwise it is an affine subspace. $U_i \in \mathbb{R}^{D \times d_i}$ is a basis for subspace S_i, and $y \in \mathbb{R}^{d_i}$ is a low dimensional representation for point $\mathbf{x}$. μ_i, U_i, n, d_i can be called as model parameters. The goal of the subspace clustering is to estimate model parameters and data segmentation which is determination of which point belongs to which subspace.

When the number of subspace is reduced to one the it is known as Principal Component Analysis(PCA). PCA can be solved easily using SVD. On the other hand, when number of the subspaces is more than one, there are some difficulties to be handled.

- First difficulty is to have strong coupling between model estimation and data segmentation. If data segmentation is given, then we can apply PCA for each cluster and obtain the subspaces or if data model is given then one could easily find data segmentation by assigning each data point to closest subspace. In general, we don't have data model and data segmentation so we need to solve them simultaneously.

- Secondly, the distribution of data is unknown. If we know that data is clustered around some centers and each centers are far way from each other then the problem reduces to well-studied central clustering problem. However, when data is distributed on the subspace which means that points from same subspace are located very far away from each other then problem becomes much more challenging.

- Thirdly, the relative position and orientation of subspaces are arbitrary. When subspaces are disjoint, which means that only intersection is the origin, and independent, which means that dimension of sum of subspaces is equal to sum of individual dimensions, the problem is easy. When there are dependent subspaces, the problem becomes more challenging.

- Fourth challenge is noisy data, missing entries, and outliers. When data is corrupted by noise, data points are not exactly on subspaces which creates difficulty. Generally speaking, noise for the case of multiple subspaces is not well-studied.

- The fifth one is model selection. Model selection is tricky, one can fit one subspace per data point or a single subspace for all data points. Both models are clearly not successful. The challenge is to find the most accurate model that have small number of subspaces with small dimensions. Generally, model selection techniques precede subspace clustering algorithms.

3. SUBSPACE CLUSTERING ALGORITHMS

Subspace clustering algorithms will be discussed very shortly in four categories. For further discussions and more details, one can read [1].

3.1. Algebraic Methods

Algebraic methods use either linear algebra or polynomial algebra. Linear algebra based methods try to apply matrix decomposition while GPCA tries to fit polynomials and try to cluster derivative vectors. Even though, these algorithms operate under noise-free assumption, they provide great insights into the geometry and algebra of the subspace clustering.

3.2. Iterative Methods

Iterative methods are generally combined with algebraic methods in order to improve the performance of algebraic methods in the case of noisy data. Main idea is to use iterative refinement. Given an initial data segmentation, one can estimate model parameters. Then given data model, one can assign data points to closest subspace. Iteratively, these methods try to converge to a better local minimum.

3.3. Statistical Methods

The other types of algorithms are not optimal in a maximum likelihood sense. In statistical methods, there is a probabilistic generative model for the data and they try to be optimal in maximum likelihood sense. Some statistical methods are mixture of probabilistic PCA, agglomerative lossy compression(ALC), random sample consensus(RANSAC).

3.4. Spectral Clustering-Based Methods

These methods start with constructing an affinity matrix. Similarity measure between data points can be distance between the points or some kind of angle between data points. One of the critical issue is to define good affinity matrix. After constructing affinity matrix, these algorithms generally continue with obtaining another matrix based on affinity matrix and apply eigendecomposition and K-means algorithms to cluster data points. Some examples of spectral clustering-based

methods are spectral local best-fits flats(SLBF), sparse subspace clustering(SSC), spectral curvature clustering(SCC).

4. APPLICATIONS IN COMPUTER VISION

There are two applications from computer vision in [1]. The first example is motion segmentation from feature points trajectories. The second example is face clustering under varying illumination.

4.1. Motion Segmentation from Feature Point Trajectories

Motion segmentation refers to problem of separating different moving objects in a video. In other words, it is to identify different spatiotemperal regions in sequence of images that corresponds to different moving objects. The algorithms start with extracting feature points from images such as edges or corners. Then the problem becomes to cluster feature point trajectories of different moving objects. The idea is that trajectories form subspaces if they are from the same moving object and one can use subspace clustering algorithms in order to do motion segmentation. [1] presents result of motion segmentation based on the dataset of Hopkins155 motion segmentation database can be seen from Fig 2 which is available at http://www.vision.jhu.edu/data/hopinks155.

The mathematical model that describes motion of feature point trajectories depends on camera projection model which is considered as affine model. Then all trajectories from the same moving object live in three dimensional affine subspace. Let $\{x_{fj} \in \mathbb{R}\}_{j=1,\dots,N}^{f=1,\dots,F}$ denote two dimensional projections of $\{X_j \in \mathbb{R}^3\}_{j=1}^{N}$ which are 3D points on a moving object in F frames and $A_f \in \mathbb{R}^{2\times 4}$ be motion matrix of frame f.

$$x_{fj} = A_f \begin{bmatrix} X_f \\ 1 \end{bmatrix}$$

Then we can stack all F tracked feature points and we get the following:

$$\begin{bmatrix} x_{11} & \dots & x_{1N} \\ . & & \\ . & & \\ . & & \\ x_{F1} & \dots & x_{FN} \end{bmatrix}_{2F\times N} = \begin{bmatrix} A_1 \\ . \\ . \\ . \\ A_F \end{bmatrix}_{2F\times 4} \begin{bmatrix} X_1 & \dots & X_N \\ 1 & \dots & 1 \end{bmatrix}_{4\times N}$$

By assuming that we are given N trajectories of n different moving objects, we know that these trajectories lie in a union of affine subspaces in $\mathbb{R}^{2F}$. Then motion segmentation is the task of clustering N points into n affine subspaces. Further discussions and results can be found in [1].

Fig. 2. Sample Images from Motion Segmentaion Dataset

4.2. Face Clustering Under Varying Illumination

The face clustering problem refers to clustering the images according to identity of the person. It has been shown by [2] that the set of all images taken under all lighting conditions can be well approximated by low-dimensional subspaces. Therefore face clustering problem is to cluster set of images according to subspaces that each subspace is equivalent to each person's identity.

n	2	3	4	5	6	7	8	9	10
GPCA	0.0	49.5	0.0	26.6	9.9	25.2	28.5	30.6	19.8
SCC	0.0	0.0	0.0	1.1	2.7	2.1	2.2	5.7	6.6
SSC	0.0	0.0	0.0	0.0	0.0	0.0	0.0	2.4	4.6
SLBF	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.2	0.9
ALC	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Table 1. Mean Percentage of Misclassification on Yale Dataset

1 shows the results of subspace clustering algorithms on Yale faces B database which is avaliable at http://cvc.yale.edu/projects/yalefacesB/yalefacesB.html. The process of experiment starts with downsampling images to 120x160. Then, PCA is used to project the images onto a subspace of dimension r=5 for GPCA and r=20 for ALC, SCC,SLBF, and SSC. Conclusions are as follows:

- GPCA does not perform well since 5 dimensions are not enough for clustering. On the other hand, increasing dimension increases the computation time exponentially.
- SSC, SSC, SLBF perform very well for face clustering.
- ALC has the best performance. However it requires to adjust a parameter δ which is very effective on the performance of ALC.

5. FACE CLUSTERING FOR PROFILE IMAGES

Some face clustering algorithms are tested on profile images. Those algorithms are GPCA(Generalized Principle

Component Analysis), SSC(Sparse Subspace Clustering), SCC(Spectral Curvature Clustering), SLBF(Spectral Local Best-fit Flats). ALC is not implemented because parameter δ couldn't be set which is very effective on the performance of ALC. Implementations of GPCA and SCC are done using Matlab code from [3]. Implementation of SLBF is done using Matlab code from [4]. Implementation of SSC is done using Matlab code from [5].

The dataset includes 107 images from 7 different persons. On average, there are 15 images from each person. Images are taken under different lighting conditions. Additionally, there are some intentional variations in dataset such as photos with glasses and without glasses. Moreover, location of the faces are not perfectly aligned throughout dataset which makes clustering harder. Some samples from dataset can be seen from Fig 3.

Before running subspace clustering algorithms, dataset is preprocessed as described in [1]. Firstly, images are down-sampled to 120x120. Then, PCA is used to project images onto subspaces with r=5 for GPCA and r=20 for SSC,SCC,SLBF. If required, dimension of subspaces are chosen as 2. Additionally, number of subspaces is provided for some algorithms when it is necessary. Moreover, for statistical algorithms, experiments are run for ten times and misclassification is calculated by taking mean average. Experiments results are given in Table 2.

n	2	3	4	5	6	7
GPCA	31.2	25.0	54.6	64.1	64.8	68.3
SCC	0	0	22.2	28.2	22.3	24.3
SSC	50.0	33.0	25.0	21.8	19.8	25.2
SLBF	0	0	6.3	24.3	20.8	27.1

Table 2. Mean Percentage of Misclassification on Profile Images

6. ACKNOWLEDGMENTS

The author thanks to Anadi Chaman, Ankit Raj, Berk Iskender, Ulas Kamaci, Sidharth Gupta and Yuqi Li for their contributions this work. They were helpful for creating the dataset. They allowed the author to take their pictures and use in this work.

Fig. 3. Samples from Dataset for Clustering of Profile Images

7. REFERENCES

[1] R. Vidal, "Subspace clustering," *IEEE Signal Processing Magazine*, vol. 28, no. 2, pp. 52–68, March 2011.

[2] J. Ho, Ming-Husang Yang, Jongwoo Lim, Kuang-Chih Lee, and D. Kriegman, "Clustering appearances of objects under varying illumination conditions," in *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, June 2003, vol. 1, pp. I–I.

[3] JHU Computer Vision Lab, "MS Windows NT kernel description," 2018.

[4] Teng Zhang, Arthur Szlam, Yi Wang, and Gilad Lerman, "Hybrid linear modeling via local best-fit flats," *CoRR*, vol. abs/1010.3460, 2010.

[5] Guangliang Chen and Gilad Lerman, "Spectral curvature clustering (scc)," *International Journal of Computer Vision*, vol. 81, no. 3, pp. 317–330, Mar 2009.