POWER SPECTRAL CLUSTERING ON HYPERSPECTRAL DATA

Aditya Challa, Sravan Danda, B S Daya Sagar, Laurent Najman

POWER SPECTRAL CLUSTERING ON HYPERSPECTRAL DATA

Aditya Challa, Sravan Danda, B.S. Daya Sagar

Systems Science and Informartics Unit,
Indian Statistical Institute,
Bangalore

Laurent Najman

Université Paris-Est,
LIGM, Equipe A3SI, ESIEE, France

ABSTRACT

Classification of remotely sensed data is an important task for many practical applications. However, it is not always possible to get the ground truth for supervised learning methods. Thus unsupervised methods form a valuable tool in such situations. Such methods are referred to as *clustering* methods. There exists several strategies for clustering the given data - K-means, density based methods, spectral clustering etc. Recently we proposed a novel method for clustering data - *Power Spectral Clustering*. In this article we aim to introduce the method in the context of Geoscience and Remote Sensing, apply the method to hyperspectral data and validate its applicability to remotely sensed images.

Index Terms— Clustering, Hyperspectral Data

1. INTRODUCTION

Hyperspectral images goes beyond the visible spectrum and records the values throughout the spectrum of reflected light. This leads to a capability of potentially distinguishing between various materials using remotely sensed data. This also means that at every pixel we have a number of spectral channels ranging from 126 to 512 depending on spectral resolution. Classification of hyperspectral images is one of the most actively researched problem in the field of Geoscience and Remote Sensing [1]. However ground truth data is not always available for supervised learning methods. Clustering methods make a valuable tool in such situations.

The clustering problem aims to find similar groups in the data without any knowledge of the ground truth, and thus is also referred to as unsupervised classification. There exists several methods for clustering data - k-means, density based methods, spectral clustering [2, 3]. K-means is a method based on distances and it cannot detect non-globular structures in the data. Density based methods are based on density calculation at a point and can detect non-globular structures. However, in high dimensions the density calculations are not reliable and hence fails for high dimensional data. Spectral clustering methods are based on distances and can also detect non globular structures in the data and is suited better for hyperspectral data. Apart from providing a simplified representation, clustering methods also provide information about the neighborhood, which in turn could be used to enhance the classification accuracy [1].

Recently in [4] the authors obtained a new method for clustering (considering the gamma limit of spectral clustering). In this article our aim is to explore it suitability in the context of Geoscience and Remote Sensing applications and in particular to that of hyperspectral images. In section 2 we explain the basic concepts of *Power Spectral* clustering, and look at its applications in section 3. We conclude with a brief overview and research in future directions in section 4.

2. THEORY

Let $\mathcal{G} = (V, E, W)$ denote an edge weighted graph, with V denoting vertices, E denoting edges and W the weight matrix. Assume that there are j distinct weights, $0 < w_1 < \cdots < w_j < 1$. D denotes the degree matrix of the graph. The *laplacian* is then defined as $L = D - W$. $\mathcal{G}^{(p)} = (V, E, W^{(p)})$ denotes the graph where every weight is raised to the power p. $\mathcal{G}_k = (V, E_k)$ denotes the graph with the edge set E_k consisting of edges whose weight equals w_k. $\mathcal{G}_{\geq k} = (V, E_{\geq k})$ denotes the graph with the edge set $E_{\geq k}$ consisting only of edges whose weight is at least w_k. One can accordingly define L_k and $L_{\geq k}$.

2.1. Spectral Clustering

The basic algorithm for spectral clustering is as follows -

1. Given the data, construct a graph $\mathcal{G}$ and calculate its laplacian, L.
2. Suppose that if m clusters are required, calculate the first m eigenvectors of the laplacian L. (ordered from smallest to largest).
3. Represent each of the points in the original dataset by a point in $\mathbb{R}^m$ consisting of the corresponding value in each of the eigenvectors.
4. In this space, use k-means or some classical algorithm to find the clusters.

There exists several explanations as to why such an algorithm finds non-globular structures in the data. One such explanation is obtained by interpreting the above algorithm in an optimization framework [5]. Consider the following cost function

$$Rcut(A_1, A_2, \cdots, A_k) = \sum_i \frac{W(A_i, \overline{A_i})}{|A_i|} \tag{1}$$

It can be shown that [5] minimizing the cost function above is approximately equivalent to solving the following optimization problem.

$$\begin{aligned} \underset{H \in \mathbb{R}^{n \times k}}{\text{minimize}} \quad & Tr(H^t L H) \\ \text{subject to} \quad & H^t H = I \end{aligned} \tag{2}$$

The solution to the optimization problem in (2) is given by taking the first k eigenvectors of the laplacian L as the columns of H [6]. This results in the spectral clustering algorithm as described before.

2.2. Power Spectral Clustering

Let $\mathcal{P}^{(p)}$ denote the optimization problem

$$\begin{aligned} \underset{H \in \mathbb{R}^{n \times k}}{\text{minimize}} \quad & Tr(H^t L^{(p)} H) \\ \text{subject to} \quad & H^t H = I \end{aligned} \tag{3}$$

We are then interested in calculating the limit

$$\lim_{p \to \infty} \arg\, \min \mathcal{P}^{(p)} \; = ? \tag{4}$$

Such a limit is referred to as the $\Gamma-$limit [7]. The technique of adding a parameter and calculating the $\Gamma-$limit was successfully used in various cases. The main advantage of such a technique is that, it allows us to obtain novel methods to solve the problem, to find a simple approximate solution to the complex problem in various scenarios or obtain efficient algorithms to calculate the solution [8, 9].

For the case of spectral clustering, algorithm 1 calculates the approximate $\Gamma-$limit. The proof of correctness and the validity of the approximation are discussed in [4].

How does the algorithm work? Firstly, given the data one has to construct the edge weighted graph, $\mathcal{G}$. Then starting with the highest level (edges of the highest weight), recursively add edges while the number of connected components are greater than the required number of clusters m. This is referred to as the *MST-phase*. Then construct an initial representation of the data points as described in step 5 of the algorithm. In the remaining part, we iteratively improve the representation with respect to the lower weight edges as described in steps 6-13.

Observe that, once the number of columns is equal to the the required number of clusters m, then the algorithm simply multiplies N with an orthogonal matrix. This implies that the representation does not meaningfully change once the number of columns in N is equal to m, and hence it is used as a stopping criterion in the algorithm.

Algorithm 1 Efficient algorithm to compute $\Gamma-$limit for ratio-cut.

Input: A weighted graph, $\mathcal{G}$, with distinct weights $w_1 < w_2 < \cdots < w_j$. Number of clusters, m.

Output: N - A representation of the subspace spanned by the $\Gamma-$limit of the minimizers.

1: Set $k = j$.
2: **while** Number of connected components of $\mathcal{G}_{\geq k}$ is greater than or equal to m **do**
3: Set $k = k - 1$ {We refer to this as an MST-Phase}
4: **end while**
5: Construct N by stacking the vectors $\mathbf{1}_{A_i}/\sqrt{|A_i|}$ in columns, where A_i is a connected component of $\mathcal{G}_{\geq k}$.
6: Set $l_1 = 0$ and $l_2 =$ number of connected components in $G_{\geq k}$
7: Consider the graph $\mathcal{G}_k$ and let L_k be the corresponding laplacian.
8: Set $C = [N^t L_k N]_{l_2, l_2}$
9: Calculate the first eigenvectors of eigenvalue problem whose eigenvalue is less than or equal to $\lambda_{(m)}$.

$$Cx = \lambda x \tag{5}$$

10: Let A be the matrix obtained by stacking the eigenvectors as columns.
11: Construct $\hat{A}$ as

$$\hat{A} = \begin{bmatrix} \mathbf{I} & 0 \\ 0 & A \end{bmatrix} \tag{6}$$

12: Update l_1 and l_2.
13: $N = N \times \hat{A}$
14: Set k = k - 1
15: **if** k = 0 **or** number of columns of N is equal to m **then**
16: **return** N
17: **else**
18: Goto Step (7)
19: **end if**

Algorithm 1 provides an alternate representation to that of spectral clustering. One can then use k-means algorithm to cluster the data using N as the representation. This procedure for clustering is called *Power Rcut* clustering. Observe that one can substitute the laplacian with normalized laplacian and repeat the procedure. This procedure is referred to as *Power Spectral* clustering.

Before considering the clustering method in the context of hyperspectral data, we briefly mention a few important points to note. Please refer to [4] for details.

1. Not all properties are preserved while considering the $\Gamma-$limit. In this case, the property of *non-trivial* clus-

ters applicable to the ratio cuts is not preserved. Thus, one has to post process the results to ignore the small clusters.

2. This method is closely related to Minimum spanning tree (MST) based clustering. In particular, if at the end of while loop in steps 2-4 in algorithm 1, we have exactly m components, then the clusters are same as these m components.

3. Unlike MST based clustering, the ties are not broken arbitrarily. Instead the cut is based on the size of the clusters.

3. APPLICATIONS

In this section we consider the application of the Power spectral clustering to the hyperspectral data. The datasets are taken from [10]. We consider 3 sample images - Indian Pines, a part of the Salinas and a part of Pavia University.

Note that clustering in the context of images is closely related to segmentation. In these datasets we are interested in segmenting the image to identify the various classes. Thus, we consider the 4-adjacency graph of the image where the edges are weighted by $\mathbb{L}_2$ norm. Also, we over segment the images for the following reasons -

1. Not all points are classified in the ground truth. This implies that there exists several gaps in the images, and thus one may not achieve connected components.

2. The main focus of the application is to identify the homogeneous regions in the image. Also, the regions can later be merged together depending on the requirements.

The results obtained by Power Rcut on the three sample images are shown in figure 1. Figure 1 (a) and (b) denote the Indian Pines dataset. Observe that the overall structure is preserved in the image. The boundaries however are not that sharp as the ground truth, since not all points in the ground truth image are classified. Another point to note is that the classes are **not** color coded, i.e each class is not denoted by the same color. Our aim is to show that our method results in acceptable segmentation of the image. Figure 1 (c) and (d) shows a part of the Salinas dataset. Observe, once again, that the overall structure is preserved. The consistency of the classes obtained is analyzed in section 3.1.

Figure 1 (e), (f) and (g) shows a part of the Pavia University dataset. Figure 1(g) shows the results obtained by spectral clustering. Observe that Power Rcut preserves the structure better that spectral clustering. In particular spectral clustering seems to prefer convex clusters more than Power Rcut. See for instance the buildings (marked in red in the ground truth image).

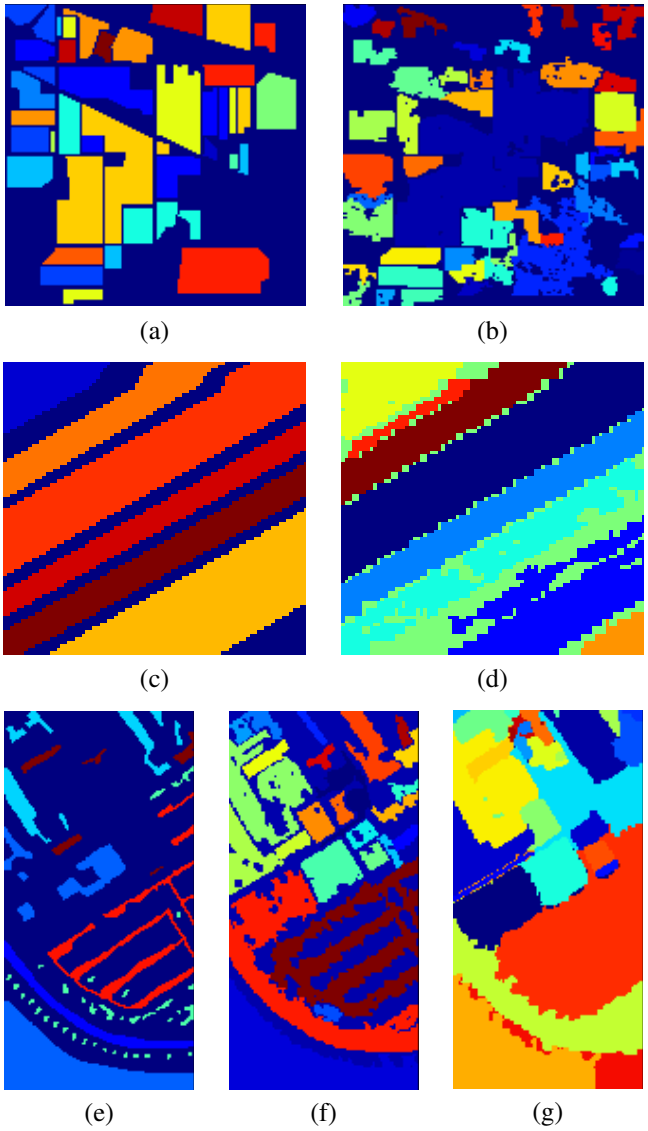

Fig. 1. (a) Indian Pines dataset - Ground Truth. (b) Indian Pines dataset - Power Rcut result. (c) part of Salinas dataset - Ground Truth. (d) part of Salinas dataset - Power Rcut result. (e) part of Pavia Univeristy dataset - Ground Truth. (f) part of Pavia Univeristy dataset - Power Rcut result. (g) part of Pavia Univeristy dataset - Spectral Clustering result.

3.1. Validation Results

To validate the results of Power Rcut, we consider *Adjusted mutual information* [11] defined by

$$AMI(U,V) = \frac{MI(U,V) - E(MI(U,V))}{max(H(U),H(V)) - E(MI(U,V))} \quad (7)$$

As mentioned earlier, we are interested in identifying the homogeneous regions in the image, and thus over segment the image. Mutual information measures the consistency of

Dataset (Method)	Score	Adjusted Score
Indian Pines (Power Rcut)	0.38	0.73
Salinas (Power Rcut)	0.66	0.89
Pavia University (Power Rcut)	0.23	0.81
Pavia University (Spectral)	0.16	0.79

Table 1. AMI scores on various datasets. Column 2 indicates the actual AMI score. Column 3 indicates the AMI score ignoring class 0

the clustering results, and is not constrained by the number of clusters being equal. Hence it is the ideal validation measure to compare with the ground truth. Adjusted mutual information adjusts for chance. This also implies that the scores are bounded above by 1, i.e a score of 1 indicates a perfect matching. See [12] for more details.

The adjusted mutual information scores for various dataset/methods is given in table 1. The second column indicates the scores. Recall that not all pixels are classified in the datasets and hence class 0 (indicating unclassified pixels) must be ignored before calculating the scores. The third column indicates the adjusted scores for the datasets. It can be deduced from the results that the Power Rcut identifies the classes well.

4. CONCLUSION AND FUTURE WORK

In this article, we introduced a new method for clustering - *Power Rcut*, and analyzed its application to the hyperspectral data. We concluded that the method identifies the homogeneous regions in the image satisfactorily. However, clustering the image is usually a first step in understanding the data. We have shown that our method can be used to visualize an unlabelled data to identify the homogeneous regions.

Another application of clustering is to improve the accuracy of classification by incorporating the spatial information as well [1]. This is considered for the future work.

5. ACKNOWLEDGEMENTS

AC and SD would like to thank Indian Statistical Institute. This work has been partly funded by ANR-15-CE40-0006 CoMeDiC and ANR-14-CE27-0001 GRAPHSIP research grants. BSDS would like to acknowledge the support received from the Science and Engineering Research Board (SERB) of the Department of Science and Technology (DST) with the grant number EMR/2015/000853, and the Indian Space Research Organization (ISRO) with the grant number ISRO/SSPO/Ch-1/2016-17.

6. REFERENCES

[1] Yuliya Tarabalka, Jón Atli Benediktsson, and Jocelyn Chanussot, "Spectral–spatial classification of hyperspectral imagery based on partitional clustering techniques," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 8, pp. 2973–2987, 2009.

[2] Anil K Jain, M Narasimha Murty, and Patrick J Flynn, "Data clustering: a review," *ACM computing surveys (CSUR)*, vol. 31, no. 3, pp. 264–323, 1999.

[3] Charu C. Aggarwal and Chandan K. Reddy, *Data Clustering: Algorithms and Applications*, Chapman & Hall/CRC, 1st edition, 2013.

[4] Aditya Challa, Sravan Danda, B S Daya Sagar, and Laurent Najman, "An Introduction to Gamma-Convergence for Spectral Clustering," Tech. Rep. hal-01427957, Jan. 2017, preprint.

[5] Ulrike Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.

[6] Helmut Lütkepohl, *Handbook of Matrices*, Wiley, 1 edition, 1997.

[7] Andrea Braides, *Gamma-convergence for Beginners*, vol. 22, Clarendon Press, 2002.

[8] Camille Couprie, Leo Grady, Laurent Najman, and Hugues Talbot, "Power watershed: A unifying graph-based optimization framework," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 7, pp. 1384–1399, 2011.

[9] Laurent Najman, "Extending the powerwatershed framework thanks to Γ-convergence," Tech. Rep. hal-01428875, Université Paris-Est, LIGM, ESIEE Paris, Jan. 2017.

[10] "Hyperspectral remote sensing scenes," `http://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes`, Accessed: 2016-12-28.

[11] "Adjusted mutual information," `http://scikit-learn.org/stable/modules/generated/sklearn.metrics.adjusted_mutual_info_score.html#sklearn.metrics.adjusted_mutual_info_score`, Accessed: 2016-12-28.

[12] Nguyen Xuan Vinh, Julien Epps, and James Bailey, "Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance," *Journal of Machine Learning Research*, vol. 11, no. Oct, pp. 2837–2854, 2010.