平成 29 年度

筑波大学情報学群情報科学類

卒業研究論文

題目 Title of the Thesis

主専攻　　ソフトウェアサイエンス主専攻

著者　　Farley Oliveira

指導教員　櫻井鉄也

Abstract

Here you write the abstract of your thesis.

Contents

List of Figures

Chap. 1 Introduction

1.1 Notation

Chap. 2 Spectral Clustering

Here we define the clustering problem, describe general ways in which it can be solved, and introduce spectral clustering as a solution to this problem which uses the variational theorem. Readers who are already familiar with the derivation of spectral clustering may feel free to skip this chapter.

2.1 The clustering problem

Clustering is the most popular way of conducting unsupervised learning currently. Given a dataset $\mathcal{D}$, the objective of clustering is to find a partition of $\mathcal{D}$, $(\mathcal{P}_1, \mathcal{P}_2, \cdots, \mathcal{P}_k)$, where $k \in \mathbb{N}^*$ is predetermined by the user of the algorithm, such that the similarity of elements of a same subset $\mathcal{P}_i$ $(i \in [\![1, k]\!])$ are as big as possible and the similarity of elements of different subsets $\mathcal{P}_i$ and $\mathcal{P}_j$ $((i, j) \in [\![1, k]\!]^2, i \neq j)$ are as small as possible. In other words, a clustering algorithm assigns a label $l \in [\![1, k]\!]$ to each data instance in $\mathcal{D}$ in such a way that data instances which are similar to each other are assigned the same label. The way the similarity of elements of a same subset and the similarity of elements of different subsets are calculated depends on the clustering algorithm used. We can say the same about the way in which the dataset $\mathcal{D}$ is represented.

Clustering may be achieved by several different approaches, each with its own advantages and disadvantages. Some models and approaches for clustering are as follows:

- Strict partioning clustering: each data instance is classified into one cluster based on its similarity with other instances. The main approach for this type of clustering is k-means: the algorithm works by iteratively assigning a label to each data instance based on its similarity with each cluster. Here, the similarity of a data instance and a cluster is obtained by computing the similarity between the instance and some kind of representative data instance of the cluster, usually some kind of mean vector.

- Hierarchical clustering: the data is divided in clusters which make up a hierarchy. This type of clustering may be achieved by two main approaches: the agglomerative approach, where each data instance starts in its own cluster, and pairs of clusters are merged as we go up in the hierarchy; and the divisive approach, where all data instances start in a same cluster, and clusters are split as we go down the hierarchy. One advantage of hierarchical clustering is that the algorithm user does not need to set the number of subsets k ahead of time.

- Overlapping clustering: In the final result, each data instance may be an element of more than one

cluster. In other words, $(\mathcal{P}_1, \mathcal{P}_2, \cdots, \mathcal{P}_k)$ is not necessarily a partition of $\mathcal{D}$. This approach may be useful when certain data instances naturally pertain to more than one class.

In contrast to the approaches above, spectral clustering works by transforming the data into a graph, constructing a certain matrix associated to this graph called Laplacian, computing the eigenvalues and eigenvectors of the Laplacian, and finally using this eigeninformation to classify the data. Although spectral clustering is often more difficult to implement (requiring, e.g., an algorithm to efficiently solve an eigenproblem), it is more general than the more common approaches such as k-means and hierarchial clustering. This is because spectral clustering may be succesfuly used for data that are arranged in complex shapes (as long as each cluster is connected), since the data is first mapped from their native data space to another one in which connectivity is preserved but geometrical relationships are simplified.

There are two main approaches with which spectral clustering can be derived. The first approach, the *ideal case* approach, considers regular Laplacian matrices as perturbations of an ideal case in which data points that are to be classified into different clusters are infinitely far apart. The second approach, the *relaxation* approach, considers spectral clustering as a approximation algorithm to solve a original NP-hard discrete optimization problem. The first approach is related to the FAST-GE2 algorithm, and the second one is related to the Bethe Hessian spectral clustering algorithm, both of which will be discussed henceforth in this thesis. For this reason, we will explain both approaches in this chapter.

2.2 Preliminary definitions

Before entering in the discussion of each derivation, we will give some definitions common to both approaches. Here we will assume that each data instance $d \in \mathcal{D}$ is a n dimensional column vector, i.e. $\mathcal{D} \subset \mathbb{R}^n$.

Definition Let $\mathcal{D}$ be a dataset containing m elements. The *similarity matrix* $A \in \mathbb{R}^{m \times m}$ associated with $\mathcal{D}$ is defined as follows:

$$A_{ij} = s(d_i, d_j), \text{ for each } (i, j) \in [\![1, m]\!]^2, \tag{2.1}$$

where s is a similarity measure and $d_i \in \mathcal{D}$ for each $i \in [\![1, m]\!]$. In this thesis, we will mainly use the *Gaussian similarity measure*, which is given by

$$s(x, y) = \exp\left(-\frac{1}{2\sigma^2}\|x - y\|^2\right), \tag{2.2}$$

where x and y are elements of a normed vector space and $\sigma \in \mathbb{R}$ is a parameter set by the user which controls the width of the neighborhoods.

We can think of the similarity matrix above as encoding a weighted graph $G_{\mathcal{D}} = (V_{\mathcal{D}}, E_{\mathcal{D}})$ representing the dataset $\mathcal{D}$. In this case, each element A_{ij} of A represents the weight of an edge connecting the vertices $(v_i, v_j) \in {V_{\mathcal{D}}}^2$.

Definition Let $A \in \mathbb{R}^{m\times m}$ be a similarity matrix and $G_{\mathcal{D}} = (V_{\mathcal{D}}, E_{\mathcal{D}})$ be the graph associated with a dataset $\mathcal{D}$. The *unnormalized Laplacian matrix* of the graph $G_{\mathcal{D}}$ is defined by

$$L_0 = D - A, \tag{2.3}$$

where $D \in \mathbb{R}^{m\times m}$ is defined to be the diagonal matrix whose D_{ii} ($i \in [\![1, m]\!]$) elements are given by the sum of the elements of the matrix A's i-th row.

Definition Let $L_0 \in \mathbb{R}^{m\times m}$ be the unnormalized Laplacian matrix of, and $G_{\mathcal{D}} = (V_{\mathcal{D}}, E_{\mathcal{D}})$ be the graph associated with, a dataset $\mathcal{D}$. The *normalized Laplacian matrix* of the graph $G_{\mathcal{D}}$ is defined by

$$L = D^{-1/2} L_0 D^{-1/2}, \tag{2.4}$$

where $D \in \mathbb{R}^{m\times m}$ is defined to be the diagonal matrix whose D_{ii} ($i \in [\![1, m]\!]$) elements are given by the sum of the elements of the matrix A's i-th row.

2.3 The ideal case approach

Here we will consider the ideal case for spectral clustering where $A_{ij} = 0$ whenever d_i and d_j ($(i, j) \in [\![1, m]\!]^2$) are in different clusters and $A_{ij} > 0$ otherwise. For convenience, we will only consider the case where the number of clusters k is 3 and we will assume that $d_i \in \mathcal{D}$, for $i \in [\![1, m]\!]$, are ordered in such a way that $i < j$ whenever the label of d_i is smaller than the label of d_j (remember that the labels l are elements of $[\![1, k]\!]$). The argument, however, can be easily extended to a general case.

In this section, for algebraic convenience, we will use an alternative definition for the unnormalized Laplacian matrix: $L_{\text{new}} = D^{-1/2} A D^{-1/2}$, instead of $L = D^{-1/2} L_0 D^{-1/2}$. The use of this trick is justified by the fact that, since $L_0 = D - A$, we have that $L_{\text{new}} + L = I_m$, where I_m is the identity matrix of order m. Therefore L_{new} and L possess the same eigenvectors, and, for each $i \in [\![1, m]\!]$, if λ_i is an eigenvalue of L, then $1 - \lambda_i$ is an eigenvalue of L_{new}. Outside of this section, however, we will use the normal definition of unnormalized Laplacian as given in the previous section.

Before entering the derivation, we need to outline a result.

Proposition 2.3.1 *Let $\mathcal{D}$ be a dataset such that its associated graph $G_{\mathcal{D}}$ is connected. Then the normalized Laplacian associated with $\mathcal{D}$ has the eigenvalue* 1 *with positive eigenvector. Furthermore, all the other eigenvalues are strictly smaller than* 1.

This is a basic result in spectral graph theory. A proof may be found in, e.g., [1].

Since $A_{ij} = 0$ whenever d_i and d_j are in different clusters, $A \in \mathbb{R}^{m\times m}$ (where m is the number of data instances in $\mathcal{D}$) may be expressed as a block matrix as follows:

$$A = \begin{bmatrix} A^{(1)} & 0_{m_1\times m_2} & 0_{m_1\times m_3} \\ 0_{m_2\times m_1} & A^{(2)} & 0_{m_2\times m_3} \\ 0_{m_3\times m_1} & 0_{m_3\times m_2} & A^{(3)} \end{bmatrix}. \tag{2.5}$$

Here, all the elements of the three matrices $A^{(1)} \in \mathbb{R}^{m_1 \times m_1}$, $A^{(2)} \in \mathbb{R}^{m_2 \times m_2}$ and $A^{(3)} \in \mathbb{R}^{m_3 \times m_3}$ are strictly positive (that is, all their elements are strictly positive) and we have that $m_1 + m_2 + m_3 = m$. From now on, to avoid verbosity, we will omit the subscripts of the 0 matrices.

It is easy to see from the definitions that the normalized Laplacian L and the diagonal matrix D can be expressed as block matrices in a similar way:

$$D = \begin{bmatrix} D^{(1)} & 0 & 0 \\ 0 & D^{(2)} & 0 \\ 0 & 0 & D^{(3)} \end{bmatrix} \quad \text{and} \quad L = \begin{bmatrix} L^{(1)} & 0 & 0 \\ 0 & L^{(2)} & 0 \\ 0 & 0 & L^{(3)} \end{bmatrix}, \tag{2.6}$$

where $D^{(1)} \in \mathbb{R}^{m_1 \times m_1}$, $D^{(2)} \in \mathbb{R}^{m_2 \times m_2}$ and $D^{(3)} \in \mathbb{R}^{m_3 \times m_3}$ are themselves diagonal matrices and $L^{(1)} \in \mathbb{R}^{m_1 \times m_1}$, $L^{(2)} \in \mathbb{R}^{m_2 \times m_2}$ and $L^{(3)} \in \mathbb{R}^{m_3 \times m_3}$ are strictly positive normalized Laplacians of each element of the partition $(\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3)$. Here, the following relation holds for each $i \in \{1, 2, 3\}$:

$$L^{(i)} = \left(D^{(i)}\right)^{-1/2} A^{(i)} \left(D^{(i)}\right)^{-1/2}. \tag{2.7}$$

Since the matrix L is block-diagonal, its set of eigenvalues σ_L is given by the union of the set of eigenvalues of each block L_1, L_2 and L_3, respectively σ_{L_1}, σ_{L_2} and σ_{L_3}. Furthermore its eigenvectors are the same as the ones of L_1, L_2 and L_3, provided that they are "extended" with 0 elements as necessary. The proof of this claim may also be found in [1].

By proposition 2.3.1, we know that each $L^{(i)}$ ($i \in \{1, 2, 3\}$) has 1 as an eigenvalue with positive eigenvector, which we denote by $x_1^{(i)} \in {\mathbb{R}_{>0}}^{m_i}$. Furthermore, all other eigenvalues of each $L^{(i)}$ are smaller than 1. This implies that L has 1 as an eigenvalue with multiplicity 3. Let X be the matrix containing the eigenvectors associated with these eigenvalues. We have that

$$X = \begin{bmatrix} x_1^{(1)} & 0 & 0 \\ 0 & x_1^{(2)} & 0 \\ 0 & 0 & x_1^{(3)} \end{bmatrix} \in \mathbb{R}^{m \times 3}. \tag{2.8}$$

However, from elementary linear algebra, we know that for a Hermitian matrix if v_1 and v_2 are two eigenvectors associated with a certain eigenvalue, so is $\alpha v_1 + \beta v_2$, with $(\alpha, \beta) \in \mathbb{R}^2$. Since the normalized Laplacian is Hermitian, we could have picked any other 3 eigenvectors spanning the same subspace as the ones above. The actual eigenvectors we obtain may depend on the small perturbations in the normalized Laplacian and the eigensolver used. This means that we could have gotten XR instead of X, for any orthogonal matrix $R \in \mathbb{R}^{3 \times 3}$. Therefore, we make the transformation $X \longmapsto XR$ to the matrix above in our analysis.

By normalizing the rows of the matrix X, we construct the matrix $Y \in \mathbb{R}^{m \times 3}$ as follows:

$$Y = \begin{bmatrix} 1_{m_1 \times 1} & 0 & 0 \\ 0 & 1_{m_2 \times 1} & 0 \\ 0 & 0 & 1_{m_3 \times 1} \end{bmatrix} R. \tag{2.9}$$

If we let ${R_1}^T \in \mathbb{R}^{1\times 3}$, ${R_2}^T \in \mathbb{R}^{1\times 3}$ and ${R_3}^T \in \mathbb{R}^{1\times 3}$ represent the rows of the matrix R, equation (2.9) tells us that the i-th row of Y is given by ${R_j}^T$, where $i \in [\![1, m]\!]$, $j \in \{1, 2, 3\}$ and $d_i \in \mathcal{P}_j$ (i.e. the label of the i-th data instance is j).

As a result, the rows of the matrix Y related to the same label i will cluster in the same point ${R_i}^T$. Furthermore, from the fact that R is an orthogonal matrix, we deduce that rows of Y corresponding to different labels will cluster in points (in the unit sphere) that are perpendicular to each other. This permits us to use the rows of the matrix Y to easily recover the labels of each $d_i \in \mathcal{D}$, with $i \in [\![1, m]\!]$, by e.g. applying the k-means algorithm to these rows.

Needless to say, most matrices we deal with are not in the ideal form we assumed they were in this section's discussion. However, we can think of a general matrix A as being of the shape $A = A_{\text{ideal}} + E$, where A_{ideal} is a matrix in the ideal form we discussed in this section and E represents the perturbation from the ideal case. As long as the norm of E is small enough, it is possible to prove that a spectral algorithm based on the approach of this section works. A more detailed description of the approach used in this section can be found in [2].

2.4 The relaxation approach

In this section we will derive the same spectral clustering as the one in the last section by framing the clustering problem as a discrete optimization problem and relaxing it so it is not discrete anymore. Before doing that, however, we need to give some definitions and outline some preliminary results.

Definition Let $G = (V, E)$ be a graph with adjacency matrix $A \in \mathbb{R}^{m\times m}$, where $m = |V|$, and C be a subset of the set of vertices V. Set $\overline{C} = V \setminus C$. The *cut* of the subset C is defined as follows:

$$\text{cut}\,(C) = \sum_{\substack{v_i \in C \\ v_j \notin \overline{C}}} A_{ij}. \tag{2.10}$$

The cut of a set of vertices C is a measure of how much the elements of C are connected with the vertices of $\overline{C}$. For that reason, it is minimized when C is a separated component. One may try, then, to conduct clustering by minimizing the cut of the several elements of a partition of V. However, a problem with this idea is that an eventual algorithm trying to achieve this objective might minimize the cut by separating individual vertices from the rest of the graph, which is not what we desire. A possible approach to deal with this complication is to normalize the cut in such a way that small clusters are "penalized". This leads to the next definition:

Definition Let $G = (V, E)$ be a graph with adjacency matrix $A \in \mathbb{R}^{m\times m}$, where $m = |V|$, and let $(C_1, C_2, \cdots, C_k)$, where $k \in [\![1, m]\!]$, be a partition of the set of vertices V. The *RatioCut* of the partition $(C_1, C_2, \cdots, C_k)$ is defined as the following quantity:

$$\text{RatioCut}\,(C_1, C_2, \cdots, C_k) = \sum_{i=1}^{k} \frac{\text{cut}\,(C_i)}{|C_i|}. \tag{2.11}$$

In the following propositions, we will show a useful form for expressions of the type $x^T Lx$, where L is a Laplacian matrix and x is a indicator vector. As we will see later, this will come handy when we want to find relationships between Laplacian matrices and the RatioCut of certain partitions.

Proposition 2.4.1 *Let $G = (V, E)$ be a graph with adjacency matrix $A \in \mathbb{R}^{m \times m}$, where $m = |V|$, L_0 be the unnormalized Laplacian matrix associated with G, and $x \in \mathbb{R}^m$ be a real vector. Then*

$$x^T L_0 x = \frac{1}{2} \sum_{i,j=1}^{m} A_{ij}(x_i - x_j)^2. \tag{2.12}$$

Proof

$$\begin{aligned}
x^T L_0 x &= x^T D x - x^T A x \\
&= \sum_{i=1}^{m} {x_i}^2 D_{ii} - \sum_{i,j=1}^{m} x_i x_j A_{ij} \\
&= \frac{1}{2}\left(\sum_{i=1}^{m} {x_i}^2 D_{ii} - 2 \sum_{i,j=1}^{m} x_i x_j A_{ij} + \sum_{i=1}^{m} {x_i}^2 D_{ii} \right) \\
&= \frac{1}{2}\left(\sum_{i=1}^{m} {x_i}^2 \left(\sum_{j=1}^{m} A_{ij} \right) - 2 \sum_{i,j=1}^{m} x_i x_j A_{ij} + \sum_{i=1}^{m} {x_i}^2 \left(\sum_{j=1}^{m} A_{ij} \right) \right) \\
&= \frac{1}{2}\left(\sum_{i,j=1}^{m} (x_i^2 - 2x_i x_j + x_j^2) A_{ij} \right) \\
&= \frac{1}{2} \sum_{i,j=1}^{m} A_{ij}(x_i - x_j)^2. \quad \blacksquare
\end{aligned}$$

Chap. 3 Spectral Clustering with the Bethe Hessian

Chap. 4 Constrained Spectral Clustering with FAST-GE2

Chap. 5 Proposed Method

Chap. 6 Experiments

Acknowledgements

References

[1] M. W. Mahoney. Lecture Notes on Spectral Graph Methods. *ArXiv e-prints*, August 2016.

[2] Andrew Y. Ng, Michael I. Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, pages 849–856. MIT Press, 2002.