LABIN : Balanced Min Cut for Large Scale Data

OTML PROJECT REPORT

Report by:

Group 3:

CS22B2009

CS22B2010

CS22B2011

CS22B2012

---

1 Introduction

1.1 Abstract

- This report conveys our understanding of the topic "Large Scale Balanced Minimum Cut" as a group. Many spectral clustering algorithms proposed in recent decades are not scalable to large-scale data due to high computational complexities. In this paper, we propose a novel spectral clustering method for large-scale data, LABIN.
- The study was undertaken by reading and understanding the processes and methods discussed to partition large-scale data into a number of clusters. External sources we utilized to gain some of the pre-requisite knowledge related to the topic.
- Some of the main finding about the research paper were how, the traditional clustering methods do a costly job in trying to cluster large-scale data and how researchers have been developing new and improved methods in order to cluster large-scale data efficiently.
- The significance of the newfound algorithm is obtaining the capability to analyse grid-like data such as images efficiently.

We address the challenges faced in performing spectral clustering on large scale data and how LABIN optimizes both time and space complexity in the clustering process.

Clustering identifies patterns or groupings within a data-set. It's a fundamental research area, but the high computational complexity of many algorithms hinders their scalability.

Spectral clustering methods often perform eigen-decomposition of the similarity matrix and then use k-means or spectral rotation to obtain final clustering results from eigenvectors.

Modern Spectral Clustering methods can be broadly classified into two categories:

- Approximation Based
- Non-Approximation Based

Approximation based methods involve sampling, relaxation, etc., which we will discuss further.

Non-Approximation based methods are pure spectral clustering algorithms applied to the whole data-set.

This leads to the question: how can clustering be done without applying the algorithm to the whole data-set? The answer lies in the processes involved in **Approximation Based Methods**.

2 Problem Formulation

Let $X = \{x_1, x_2, \ldots, x_n\}$ be a set of data points in R^d. Consider a graph $G(X) = (X, E)$ with elements of X as vertices and undirected edges weighted by similarity between data pairs. The similarity relates to their spatial relationship, where closer points have higher similarity. The information in $G(X)$ is represented by the Affinity matrix $A(X) \in R^{n \times n}$, with i, j element being the similarity between x_i and x_j.

For a data-set of size n, with d features, traditional Spectral Clustering methods have a high time complexity of $O(n^3)$, mainly due to eigen-decomposition.

2.1 Eigen Decomposition

Eigen decomposition involves decomposing a square matrix into its constituent parts. For a square matrix A of size $n \times n$, if there exists a scalar λ and a non-zero vector v such that:

$$Av = \lambda v \tag{1}$$

where v is the eigen-vector and λ is the corresponding eigen-value.

The result of the eigen-decomposition yields:

$$A = V \Sigma V^{-1} \tag{2}$$

where V is a matrix whose columns are composed of the eigenvectors of A.

2.2 Basic Approximation based methods

- Nyström method : involves accelerating the eigen-decomposition by randomly sampling a small sub-matrix from the original matrix and compute an approximation of the eigenvectors of the original matrix.
- The second type involves the usage of pre-processing to reduce the data size and perform clustering on the reduced data. This method is known as the k-means-based approximate spectral clustering (KASP) method, which employs k-means to partition the data set into a large number of clusters and then partition these cluster centers with the traditional spectral clustering.
- The third type is to use anchor-based strategy to construct a bipartite similarity matrix and perform clustering on the bipartite similarity matrix instead of the full similarity matrix.
- Landmarks-based spectral clustering (LSC), which computes a bipartite similarity matrix instead of the full similarity matrix, and then performs the eigen-decomposition on the low-size similarity matrix and the final clustering result is obtained by clustering the eigenvectors with k-means.

2.3 Spectral Clustering and Rotation

Given an affinity matrix $A \in Rn \times n$ constructed from data-set $X = x_1, \ldots, x_n$. Let $Y \in \Phi^{n \times c}$ be the cluster indicator matrix.

Our goal is to find a solution to the cluster indicator matrix. Since we are focusing on k-ways partitioning, let's look at the k-way NCut problem.

$$\min_{Y^T DY} \mathrm{Tr}(Y^T L_A Y) \tag{3}$$

where L_A is the laplacian matrix of matrix A.

A new spectral clustering method was proposed which aims to learn the connected probability between x_i and x_j where a constraint rank is imposed on the laplacian matrix of S such that the number of connected components is exactly equal to the number of clusters.

A new method was proposed where spectral-rotation was proposed to solve the following k-way NCut problem:

$$\max_{Y \in \Phi^{n \times c}, Z = Y(Y^T D_A Y)^{-\frac{1}{2}}} \mathrm{Tr}(Z^T A Z) \tag{4}$$

where Z is the scaled-partition matrix. We observe that the greatest problem arises with providing a direct solution to the above problem. So, a commonly used method to solve this problem is to relax(replace) the continuous values of the matrix Z to discrete values. The optimal solution of Z is $Z^* R : R^T R = I_c$ where $Z*$ consists of c column vectors of the eigen-vectors of $D^{-1}A$ which correspond to the c biggest eigen-values.

2.4 Anchor-Based Spectral Clustering

One of the effective solutions proposed to solve the scalability problem was using an anchor based approach which can also be called landmarks based method. The anchor-based strategy first seeks m anchors to represent the n objects, and then constructs a sample-anchor similarity matrix by calculating the distance between the anchors and objects with the constraint that ($m << n$). There are mainly two methods for anchor generation, i.e., random selection and k-means generation. K-means is preferred for anchor generation since cluster centers have a stronger representation power than random selected data.

After we have obtained m anchors $W \in R^{d \times m}$, the next step is to compute a sample-anchor similarity matrix $B \in R^{n \times m}$ such that B1 = 1. With B, we can obtain an affinity matrix A as:

$$A = B\Delta B^T A = PP^T P = B\Delta^{\frac{-1}{2}} \tag{5}$$

where $\Delta \in R^{m \times m}$ is a diagonal matrix and the j^{th} diagonal entry is defined as $\Delta_{jj} = \sum_{i=1}^{n} b_{ij}$

a method to obtain a sparse B was proposed which was to compute the anchor similarity matrix as follows:

$$b_{ij} = \frac{d_{i,k+1} - ||x_i - w_j||^2}{k d_{i,k+1} - \sum_{h=1}^{k} di, h} \tag{6}$$

It was further proved that the full affinity matrix obtained from B is actually symmetric, with this method, they have improved the SNC algorithm.

2.5 Self Balanced Minimum Cut

SBMC algorithm is capable to simultaneously minimize the graph cut and balance the partition across all clusters. In their method, they have proposed the following objective function:

$$\min_{Y \in \Psi^{n \times c}} ||A - sYY^T||_F^2 \tag{7}$$

where s¿0 is the balance parameter and Y is the cluster indicator matrix.

However, to solve the problem effectively the form of the objective function was altered like:

$$\max_{Y \in \Psi^{n \times c}, s>0} \left(2s \operatorname{Tr}(Y^T A Y) - s^2 ||Y||_b\right) \tag{8}$$

where $||Y||_b$ is the balance regularization used to obtain balanced partition.

3 LABIN : Balanced Min Cut For Large Scale Data

Suppose the objective is to cluster a data-set X into c clusters and let $Y \in \Psi^{n \times c}$ be the cluster-indicator matrix. First we construct a full affinity matrix using the anchor similarity matrix and then trying to optimize this new objective function:

$$\max_{Y \in \Phi^{n \times c}, s>0} 2s \operatorname{Tr}(Y^T P P^T Y) - s^2 ||Y||_b \tag{9}$$

The researchers have arrived at two different pathways to solve this objective function:

- Solving for Y with s fixed.
- Solving for s with Y fixed.

3.1 Singular Value Decomposition

The optimal solution of Y is $Y = Y * R : R^T R = I_c$ where Y* is the c column vectors of the eigen-vectors of Θ which correspond to the c biggest eigen-values.
It is a matrix factorization method that breaks down a matrix A into three constituent matrices, U, Σ and V^T such that:

$$A = U \Sigma V^T \tag{10}$$

U and V are orthogonal matrices and Σ is the diagonal matrix containing the singular values of A.

- The columns of U are the eigen-vectors of AA^T.
- The columns of V are the eigen-vectors of $A^T A$.
- The diagonal elements of Σ are the square roots of the eigen-values of AA^T or $A^T A$.

3.2 Solving Y with s fixed

Since we are fixing s, the problem is to be re-written as:

$$\max_{Y \in \Phi^{n \times c}} \operatorname{Tr}(Y^T \Theta Y) \tag{11}$$

where $\Theta = PP^T - \frac{s}{2} \times 1_n 1_n^T$.

According to the previous methods, the above equation can be solved by ALM, but we know that ALM performs poorly on large data-sets. If we consider the solution provided by using spectral rotation and matrix relaxation, the optimal solution of Y is $Y^* R : R^T R = I_c$. $Y^* \in \mathbb{R}^{n \times c}$ is the c column vectors of the eigen-vectors of Θ which correspond to the c biggest eigen-values.

Since Θ is an $n \times n$ matrix, the eigen-decomposition takes $O(n^3)$ time. So a solution is proposed which involves re-writing matrix Θ as:

$$\Theta = Q \begin{bmatrix} I_m & 0_m \\ 0_m^T & -\frac{s}{2} \end{bmatrix} Q^T \tag{12}$$

where $Q \in \mathbb{R}^{n \times m}$ is defined as $Q = [P\, 1_n]$. This method takes inspiration from the Anchor based (Landmarks Based Spectral Clustering).

Since $n \gg m$, we perform singular value decomposition on Q such that:

$$Q = U_Q \Sigma_Q V_Q^T \tag{13}$$

where $U_Q \in \mathbb{R}^{n \times (m+1)}$ and $V_Q \in \mathbb{R}^{(m+1) \times (m+1)}$ are two unitary matrices and $\Sigma_Q \in \mathbb{R}^{(m+1) \times (m+1)}$ is a diagonal matrix.

After further transformations and solving, we arrive at a solution:

$$\Theta = U_Q \Phi U_Q^T \Phi = \Sigma V_Q^T \begin{bmatrix} I_m & 0_m \\ 0_m^T & -\frac{s}{2} \end{bmatrix} V_Q \Sigma_Q \tag{14}$$

Suppose the eigen-decomposition of Φ is $\Phi = V_\Phi \Sigma_\Phi V_\Phi^{-1}$.

$$\Theta U_Q V_\Phi = U_Q V_\Phi \Sigma_\Phi Q^T Q = V_Q \Sigma^2 V_Q^T \tag{15}$$

If we observe carefully we only perform eigen-decomposition of a matrix of order much less than n and SVD of a matrix of order $n \times (m+1)$. This causes the final time complexity to be $O(nm^2)$.

3.3 Solving s with Y fixed

If Y, the cluster-indicator matrix, is fixed, the problem becomes:

$$\min_{s>0} \left(s - \frac{\mathrm{Tr}(Y^T A Y)}{||Y||_b} \right)^2 \tag{16}$$

with the optimal solution being:

$$s = \frac{\mathrm{Tr}(Y^T A Y)}{||Y||_b} \tag{17}$$

The next step is to learn a suitable orthogonal matrix R and a cluster indicator matrix Y such that $Y^* R$ is closest to Y by carrying out a solution similar to the Spectral Rotation problem.

$$\min_{Y \in \Phi^{n \times c}, R \in R^{c \times c}, Y 1_c = 1_n, R^T R = I_c} ||Y - Y^* R||_F^2 \tag{18}$$

$$||Y - Y^* R||_F^2 = 2n - 2\,\mathrm{Tr}(Y^T Y^* R). \tag{19}$$

The above problem can be re-written as:

$$\max_{Y \in \Phi^{n \times c}, R \in R^{c \times c}, Y 1_c = 1_n, R^T R = I_c} \mathrm{Tr}(Y^T Y^* R) \tag{20}$$

This also gives us two ways to solve:

- Update for R with Y fixed.
  If Y is fixed, supposing the SVD of $Y^T Y$ is $U \Sigma V^T$, we have:

$$\mathrm{Tr}(Y^T Y R) = \mathrm{Tr}(R U \Sigma V^T) = \mathrm{Tr}(\Sigma E) = \sum_{i=1}^{c} \lambda_{ii} e_{ii} \tag{21}$$

  where $E = V^T R U$ λ_{ii} and e_{ii} are the diagonal elements of Σ and E. It is verified that $\sum_{j=1}^{c} e_{ji}^2 = 1$. Since λ_{ii} was proven to be a singular non-negative value, this gives $\mathrm{Tr}(\Sigma E) = \sum_{i=1}^{c} \lambda_{ii} e_{ii} <= \sum_{i=1}^{c}$.
  The equality holds if and only if $e_{ii} = 1$.
  This means that $\mathrm{Tr}(\Sigma E)$ reaches maximum when $E = I_c$. We obtain optimal solution of R as:

$$R = V U^T \tag{22}$$

- Update for Y with R fixed.
  Let $G = Y^*R$. According to problem, the optimal solution of Y can be obtained by solving the following problem:

$$\max_{Y \in \Phi^{n \times c}, Y1_c = 1_n} \mathrm{Tr}(GY) \tag{23}$$

  which is re-written as :

$$\max_{Y \in \Phi n \times c, Y1_c = 1_n} \sum_{j=1}^{c} \sum_{i=1}^{n} y_{ij} g_{ij} \tag{24}$$

  The optimal solution of y_i was obtained as:

$$y_{ij} =< l = arg \max_{j' \in [1,c]} g_{ij'} > \tag{25}$$

  Here, this above equation gives either True or False value and $G = Y^*R$.

3.4 LABIN : The optimization Algorithm

Algorithm 1 LABIN : Balanced Min Cut for Large Scale Data

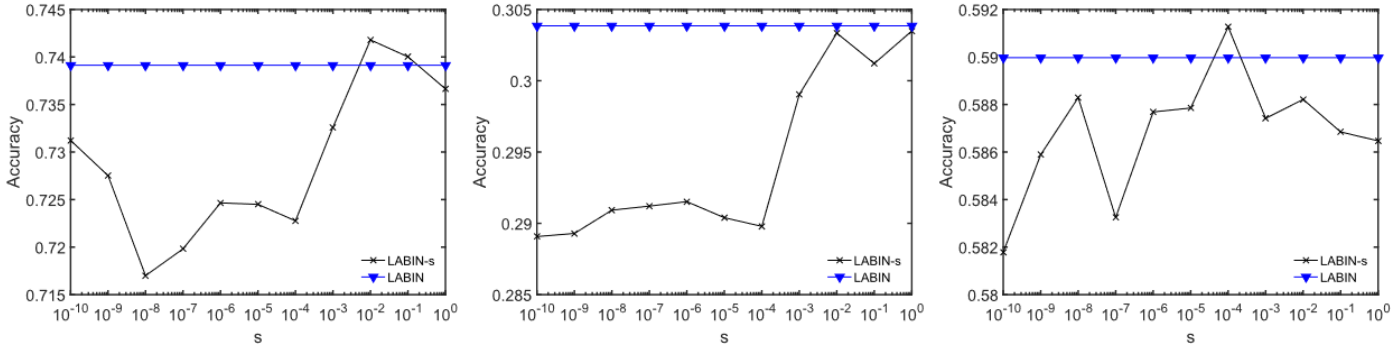
Input: Data matrix $X \in R^{d \times n}$, number of nearest neighbours k and anchors m
1: Find m anchors W with k-means and construct a sparse representation matrix $B \in R^{n \times m}$
2: Compute $P \in R^{n \times m} = B\Delta^{-\frac{1}{2}}$, where $\Delta \in R^{n \times m}$ is the degree matrix of B.
3: Perform the reduced SVD on $Q = [P, 1_n]$ such that $Q = U_Q \Sigma_Q V_Q^T$
4: Randomly initialize Y.
repeat
  Update $s = \frac{\mathrm{Tr}(Y^T P P^T Y)}{||Y||_b}$
  Update Φ accordingly and perform eigen-decomposition of Φ as $\Phi = V_\Phi \Sigma_\Phi V_\Phi^{-1}$
  Update Y^* as the c column vectors of $U_Q V_\Phi$ which correspond to the c biggest eigen-values in Σ_Φ
  repeat
    Update R according to $R = VU^T$
    Update Y according to $y_{ij} =< l = arg \max_{j' \in [1,c]} g_{ij'} >$
  until $\max_{Y \in \Phi^{n \times c}, R \in R^{c \times c}, Y1_c = 1_n, R^T R = I_c} \mathrm{Tr}(Y^T Y^* R)$ converges
until $\max_{Y \in \Phi^{n \times c}, s > 0} 2s\,\mathrm{Tr}(Y^T P P^T Y) - s^2 ||Y||_b$ converges
Output: The clustering result Y.

4 Results and Analysis

A total of 11 data-sets were taken to test the validity of the new algorithm, the testing was carried out for these data-sets by LABIN and 4 approximation based spectral clustering methods: KASP, Nyström, Landmarks based SC, Scalable Normalized Cut, along with NCut with k-means and RCut with k-means to check if LABIN can produce comparable results with non-approximate methods. Results show that LABIN:

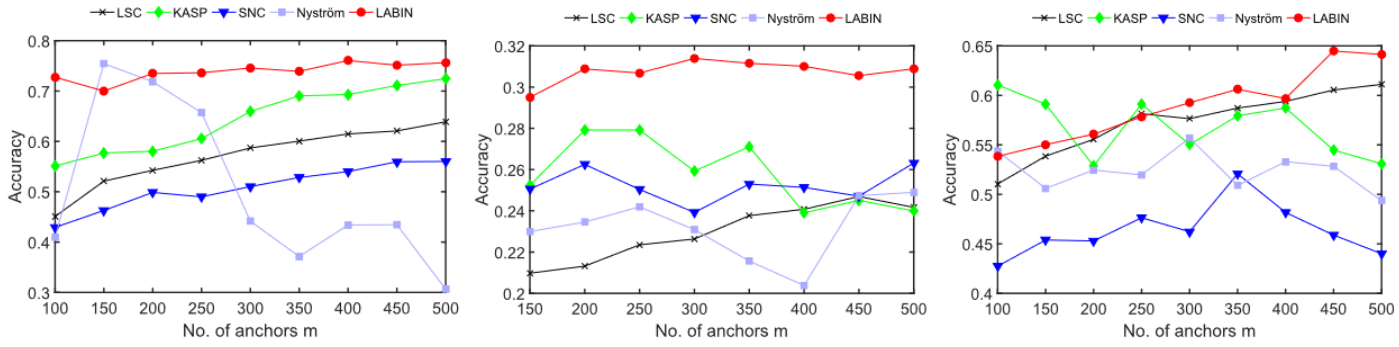
- Outperformed all four approximation based spectral clustering methods.
- Achieved 18% improvement compared to Scalable Normalized Cut in terms of accuracy.
- Showed a 18% improvement in terms of assigning the data-points to their true clusters.
- Outperformed SBMC, LABIN's parent algorithm on their performance on large-scale data-sets.

With respect to the balance parameter, it has been found through these experiments the fixing the cluster indicator matrix and then solving for s, is very time consuming.
However the clustering result obtained is closest to the optimal solution, when we fix s and Y can be obtained in at-least 1 iteration.

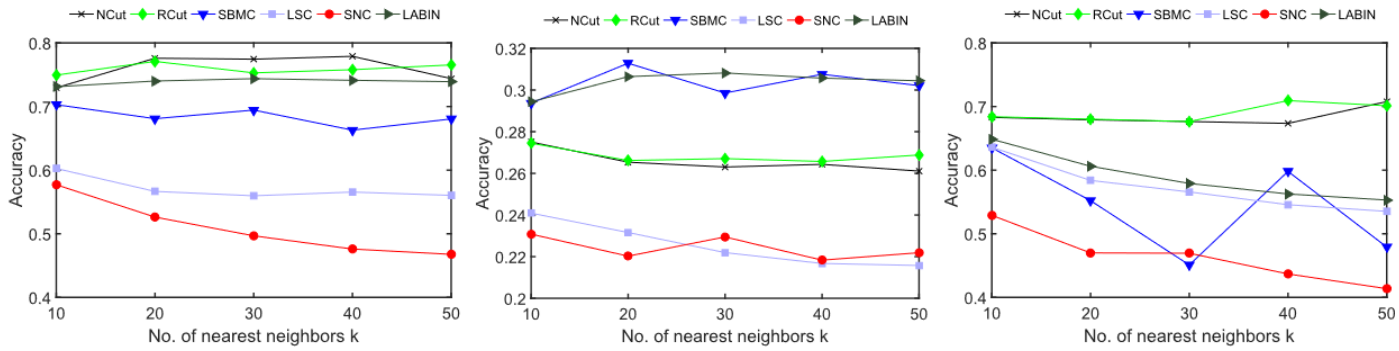


(a) (b) (c)

With increase in the number of anchors, performance of the model improves, whereas, with increase in the number of neighbours, the performance is affected negatively.



(a) (b) (c)



5 Conclusion and Comparison

To conclude our report on the research paper, we have understood the optimization process and algorithm of LABIN : Balanced Min Cut For Large Scale Data.
This section also aims to compare the stated algorithm in the research paper and the code which our group has written which was guided by the same algorithm.

- In the research paper, a new spectral clustering algorithm has been proposed for large-scale data, which involved the construction of a sample-anchor similarity matrix, instead of the full similarity matrix, in which improved Spectral Rotation methods are used to optimize the new model.
- This method is efficiently able to perform eigen-decomposition with a pseudo-linear time and space computational complexities.
- Compared with the original SBMC, this method has a significant reduction in space-complexity due to ISR(Improved spectral Rotation).
- Extensive experiments on large-scale data show the effectiveness and efficiency of LABIN compared to the conventional methods.
- As per the research paper, only flaw in LABIN is that it is a two-step optimization process.
- According to the researchers, a direct solution to the LABIN optimization problem can be obtained. But currently we donot have access to it.
- As for the algorithm, a working code was written by our tem to implement the LABIN algorithm for a random dataset which yielded the expected results.
- The implemented code was able to perform the LABIN algorithm of the data-set as per the stated time complexity constraint O(nm^2).
- As for improvements regarding the proposed algorithm, we as a team couldn't identify a potential scope for improvement.