Data Ranking and Clustering via Normalized Graph Cut Based on Asymmetric Affinity

Olexiy Kyrgyzov[1], Isabelle Bloch[1], Yuan Yang[1], Joe Wiart[2], and Antoine Souloumiac[3]

[1] Institut Mines-Telecom, ParisTech/CNRS LTCI, Whist Lab, 75013, Paris, France
{olexiy.kyrgyzov,isabelle.bloch,yuan.yang}@telecom-paristech.fr
[2] Orange Labs* R&D, Whist Lab, 92130, Issy-les-Moulineaux, France
joe.wiart@orange-ftgroup.com
[3] CEA, LIST, Laboratoire Outils Analyse de Données, 91191, Gif-sur-Yvette, France
antoine.souloumiac@cea.fr

Abstract. In this paper, we present an extension of the state-of-the-art normalized graph cut method based on asymmetry of the affinity matrix. We provide algorithms for classification and clustering problems and show how our method can improve solutions for unequal and overlapped data distributions. The proposed approaches are based on the theoretical relation between classification accuracy, mutual information and normalized graph cut. The first method requires a priori known class labeled data that can be utilized, e.g., for a calibration phase of a brain-computer interface (BCI). The second one is a hierarchical clustering method that does not involve any prior information on the dataset.

Keywords: graph cut, asymmetric affinity, mutual information, BCI.

1 Introduction

Separation of the informative part from noise or artifacts in observed data is a significant step for dimension reduction and feature extraction. The main aim of this work is to increase performances of classification algorithms for multi-class data, which may have different types of distribution, and reveal clusters based on data affinity information.

Data mining is a process of discovering patterns and their relations into observed datasets. One of the extensively used approaches for similarity analysis between data samples is spectral graph clustering [8]. Directed graphs are asymmetric structures, which are able to involve information on data diversity, e.g. an employer may know some personal data about his employee while the last one is acquainted only with public information of his boss [11].

The problem with different data distributions is the general form for natural signal analysis, e.g. speech or electroencephalogram (EEG) classification when we need to implement adaptive artifact rejection. However, existing approaches

* This work was partially supported by a grant from Orange Labs, France.

A. Petrosino (Ed.): ICIAP 2013, Part II, LNCS 8157, pp. 562–571, 2013.
© Springer-Verlag Berlin Heidelberg 2013

do not handle properly data having different distributions, e.g. two overlapped Gaussian and uniform distributions. In this paper, we present an extension of the normalized graph cut approach [12] based on asymmetric affinity matrix [11]. We provide a short numerical analysis of the existing normalized cut in regard to the number of data samples in recovered clusters, propose a new invariant one and apply it to the image segmentation problem. Experimental results on real datasets show the applicability of the method to problems of data ranking and clustering.

2 Normalized Graph Cut

A set of points in a data space can be represented as a weighted undirected graph $\mathbf{G} = \{\mathbf{S}, \mathbf{A}\}$ [14], where $\mathbf{S} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ is the set of vertices and $\mathbf{A}_{ij}, i, j = 1, \ldots, n$ are edge altitudes, and denote the similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$. The measure of similarity for a graph is usually presented in matrix form and called affinity matrix [2]. An example is as follows:

$$\mathbf{A}_{ij} = e^{-\frac{||\mathbf{x}_i - \mathbf{x}_j||_F^2}{\sigma^2}},\ i, j = 1, \ldots, n, \tag{1}$$

where the Gaussian kernel bandwidth could be estimated via the Parzen-Rosenblatt method [13]: $\sigma = 1.122\, VAR[\mathbf{x}]\, n^{-2/5}$ and $VAR[\mathbf{x}]$ is the overall variance of the data $\mathbf{x}$.

Clustering problems can be expressed as the problem of splitting a set $\mathbf{S}$ into disjoint subsets $\{\mathbf{S}_1, \mathbf{S}_2, \ldots, \mathbf{S}_m\}$ or cutting the graph $\mathbf{G}$ into m subgraphs, where the similarity among the vertices in a set $\mathbf{S}_i$ is high and the similarity between different sets $\mathbf{S}_i$ and $\mathbf{S}_j$ $(i \neq j)$ is low [12]. Let us recall a few definitions from graph theory [1]: the degree of the i^{th} vertex is the cumulative weight of its connected edges: $\mathbf{D}_{ii} = \sum_{j=1}^{n} \mathbf{A}_{ij}$; the association of set $\mathbf{S}_i$ to $\mathbf{S}$, where $\mathbf{S}_i \subseteq \mathbf{S}$, is the cumulative weight of edges: $Assoc(\mathbf{S}_i, \mathbf{S}) = \sum_{\mathbf{x}_{i'} \in \mathbf{S}_i} \sum_{\mathbf{x}_j \in \mathbf{S}} \mathbf{A}_{i'j}$; the association of set $\mathbf{S}$ to $\mathbf{S}_i$, where $\mathbf{S}_i \subseteq \mathbf{S}$, is the cumulative weight of edges: $Assoc(\mathbf{S}, \mathbf{S}_i) = \sum_{\mathbf{x}_j \in \mathbf{S}} \sum_{\mathbf{x}_{i'} \in \mathbf{S}_i} \mathbf{A}_{ji'}$; a graph cut between two sets $\mathbf{S}_i$ and $\mathbf{S}_j$ is the cumulative weight of edges that connect vertices from $\mathbf{S}_i$ to $\mathbf{S}_j$: $Cut(\mathbf{S}_i, \mathbf{S}_j) = \sum_{\mathbf{x}_{i'} \in \mathbf{S}_i; \mathbf{x}_{j'} \in \mathbf{S}_j} \mathbf{A}_{i'j'}$.

Given an affinity matrix, the graph cut clustering [12] can be solved as a block-diagonalization of the affinity matrix $\mathbf{A}$ or minimization of the graph cut $Cut(\mathbf{S}_i, \mathbf{S}_j)$. To solve this problem we can assume that if such an affinity matrix is block-diagonalized, the summation of the weights of this diagonal block matrix is maximized and the summation of the weights in the off-diagonal block matrices is minimized. To minimize $Cut(\mathbf{S}_i, \mathbf{S}_j)$ we can assign a weight vector $\mathbf{w}_i$ of length n to each cluster, $i = 1, \ldots, m$. The k^{th} element of $\mathbf{w}_i$ defines a grade of membership of the k^{th} data point to the i^{th} cluster, i.e., the larger the weight, the stronger the association. Thus, the block-diagonalized task becomes a quadratic programming problem expressed as [2,9]:

$$\hat{\mathbf{w}} = \min_{\mathbf{w}} \mathbf{w}^T \mathbf{A} \mathbf{w},\ s.t.\ \mathbf{w}^T \mathbf{w} = 1, \tag{2}$$

where the solution of (2) is the eigenvector of $\mathbf{A}$ corresponding to the largest eigenvalue λ: $\mathbf{A}\hat{\mathbf{w}} = \lambda\hat{\mathbf{w}}$. In general, to select m clusters from a dataset we perform the eigenvalue decomposition of its affinity matrix $\mathbf{A}$ and define m largest eigenvalues $\lambda_i, i = 1, \ldots, m \leq n$ that represent m clusters in the data.

The described graph cut algorithm has obstacles: if data consist of its shifted copies, we cannot split clusters due to similar eigenvalues [12] and the algorithm is inclined to cut small isolated data subsets [14], since the value of $Cut(\mathbf{S}_i, \mathbf{S}_j)$ increases with the number of edges going across the two partitioned sets and with the distance between them. These problems are classically solved via a normalized form of the graph cut approach.

Normalized graph cut for a symmetric matrix $\mathbf{A}$ is a fraction of edge connections in subgraphs to all the nodes in the graph [12]:

$$nCut(\mathbf{S}_i, \mathbf{S}_j) = \frac{Cut(\mathbf{S}_i, \mathbf{S}_j)}{Assoc(\mathbf{S}_i, \mathbf{S})} + \frac{Cut(\mathbf{S}_i, \mathbf{S}_j)}{Assoc(\mathbf{S}_j, \mathbf{S})}. \tag{3}$$

The value of (3) becomes small if we cut $\mathbf{S}_i$ and $\mathbf{S}_j$ that have few edges with low weights between them and many internal edges with high weights. So, the common graph cut approach uses a part of the affinity matrix, and the normalized graph cut uses the full matrix.

To solve (3) we assume that there is a length-n vector $\mathbf{w}$, the values of which are either 1 or $-b$ [12]. These values are used to separate vertices of the graph: if the j^{th} component of $\mathbf{w}$ is 1, then the corresponding vertex belongs to the first cluster, and if it is $-b$, the vertex belongs to the second one. In matrix notations, the minimization of $nCut(\mathbf{S}_i, \mathbf{S}_j)$, based on symmetric affinity matrix, amounts to solve:

$$\hat{\mathbf{w}} = \min_{\mathbf{w}} \frac{\mathbf{1}^T\mathbf{A}\mathbf{1} - \mathbf{w}^T\mathbf{A}\mathbf{w}}{\mathbf{1}^T\mathbf{A}\mathbf{1}}, \text{ s.t. } \mathbf{w}^T\mathbf{A}\mathbf{1} = 0. \tag{4}$$

Introducing a diagonal matrix $\mathbf{D} = diag(\mathbf{A}\mathbf{1})$, where $\mathbf{A}\mathbf{1} = \sum_{j=1}^{n} \mathbf{A}_{ij}$ and $\mathbf{1}^T\mathbf{A}\mathbf{1} = \mathbf{w}^T\mathbf{D}\mathbf{w}$, this optimization problem is rewritten as:

$$\hat{\mathbf{w}} = \min_{\mathbf{w}} \frac{\mathbf{w}^T(\mathbf{D} - \mathbf{A})\mathbf{w}}{\mathbf{w}^T\mathbf{D}\mathbf{w}}, \text{ s.t. } \mathbf{w}^T\mathbf{D}\mathbf{1} = 0. \tag{5}$$

Since (5) is the generalized Rayleigh quotient [3], we can apply the Cholesky decomposition to the diagonal matrix $\mathbf{D}$, leading to

$$\hat{\mathbf{w}} = \min_{\mathbf{w}} \frac{\mathbf{w}^T(\mathbf{D}^{-\frac{1}{2}}(\mathbf{D} - \mathbf{A})\mathbf{D}^{-\frac{1}{2}})\mathbf{w}}{\mathbf{w}^T\mathbf{w}}, \text{ s.t. } \mathbf{w}^T\mathbf{w} = 1 \tag{6}$$

and minimize it by solving a standard eigenvector problem, expressed as:

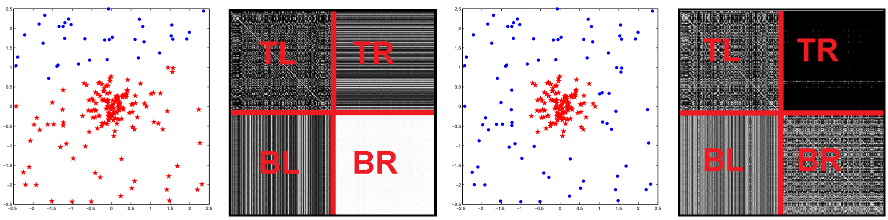
$$\mathbf{D}^{-\frac{1}{2}}(\mathbf{D} - \mathbf{A})\mathbf{D}^{-\frac{1}{2}}\mathbf{w} = \lambda\mathbf{w}. \tag{7}$$

A common approximate solution for such an integer programming problem is to compute a real vector $\mathbf{w}$ and assign its entries to $\{1, -b\}$ by testing the vector $\mathbf{w}$ against each of its entries such that the resulting partition has a minimal

$nCut(\mathbf{S}_i, \mathbf{S}_j)$ value. A critical question is also how to chose stopping criteria for $nCut(\mathbf{S}_i, \mathbf{S}j)$ that define the number of clusters in the dataset. To find eigenvectors for high-dimensional data, e.g. in image segmentation tasks, and to avoid the out of memory state, we need to take into account sparseness of the affinity matrix.

3 Normalized Graph Cut Based on Asymmetric Affinity

In this section, we present an extension of the normalized graph cut by introducing asymmetry in the affinity matrix and show its advantages for the tasks of clustering and data ranking comparing with the state-of-the-art techniques. Minimization of Eq. 3 is well adapted to datasets consisting of clusters with similar normal distributions [13]. It is based on properties of the defined symmetric affinity matrix (Eq. 1), since we use a constant σ-value for each pair of data points. An example of incorrect clustering is presented in Figure 1(a). Here we apply the normalized graph cut approach to a dataset that consists of two overlapping clusters with normal and uniform distributions. Using the symmetric affinity matrix in Figure 1(b) does not lead to a good separation, see Figure 1(a).



(a) (b) (c) (d)

Fig. 1. Overlapped data clustering based on: (a,b) symmetric and (c,d) asymmetric affinities

For real-world problems, e.g. EEG signals or biomedical image analysis, we have to deal with data that consist of overlapping clusters with various distributions. The aim of the normalized graph cut based on asymmetric affinity matrix is to involve information about cluster diversity that arises from the directed graph clustering approach [15].

To cover any type of data distribution, we use a specific kernel bandwidth for each data sample, that leads to an asymmetric form of the affinity matrix, as illustrated in Figure 1(d), and expressed as follows:

$$\mathbf{A}_{ij} = e^{-\frac{||\mathbf{x}_i - \mathbf{x}_j||_F^2}{\sigma_j^2}}, \ i, j = 1, \ldots, n, \tag{8}$$

where the kernel bandwidth, σ_j, could be evaluated as a distance to the m^{th}-neighbor, $m = \sqrt{n}$, that comes from the spacing entropy estimation [6].

We propose to evaluate the cut cost as a fraction of the total edge connections to all the nodes in the directed graph in both directions [15,7]. Due to the node directions, Eq. 9 becomes a double extension of Eq. 3. So, our new cut, which is based on the asymmetric affinity matrix, is a sum of all possible fractional combinations in the directed graph:

$$\begin{aligned} nCut(\mathbf{S}_i, \mathbf{S}_j) = & \frac{Cut(\mathbf{S}_i, \mathbf{S}_j)}{Assoc(\mathbf{S}_i, \mathbf{S})} + \frac{Cut(\mathbf{S}_i, \mathbf{S}_j)}{Assoc(\mathbf{S}, \mathbf{S}_j)} \\ & + \frac{Cut(\mathbf{S}_j, \mathbf{S}_i)}{Assoc(\mathbf{S}, \mathbf{S}_i)} + \frac{Cut(\mathbf{S}_j, \mathbf{S}_i)}{Assoc(\mathbf{S}_j, \mathbf{S}).} \end{aligned} \tag{9}$$

In Figure 1 we present the main difference between general and normalized graph cut approaches when we need to divide our graph into two subgraphs. Let us divide our affinity matrix into four pieces by two orthogonal lines (red bold lines), the cross point must be on the main diagonal and corresponds to the minimal graph cut. For the graph cut algorithm we minimize the sum of elements in the top-right square (TR), and for the normalized version we minimize

$$\frac{sum(TR)}{sum(TL) + sum(TR)} + \frac{sum(BL)}{sum(BL) + sum(BR)}, \tag{10}$$

where the letters T, B, L, R specify regions (top, bottom, left, right) of the affinity matrix and for the symmetric affinity $TR = BL^T$.

So, the common graph cut approach uses a part of the affinity matrix, and the normalized graph cut uses the full matrix. According to Figure 1, the defined normalized cut is a sum of normalized fractions:

$$\begin{aligned} & \frac{sum(TR)}{sum(TL) + sum(TR)} + \frac{sum(BL)}{sum(TL) + sum(BL)} + \\ & \frac{sum(BL)}{sum(BL) + sum(BR)} + \frac{sum(TR)}{sum(BR) + sum(TR)}, \end{aligned} \tag{11}$$

where $TR \neq BL$.

Reasoning similarly as in the previous section, the optimization problem for the normalized graph cut based on asymmetric affinity is an extension of Eq. 4, and expressed as:

$$\hat{\mathbf{w}} = \min_{\mathbf{w}} \left(2 \frac{\mathbf{1}^T \mathbf{A} \mathbf{1} - \mathbf{w}^T \mathbf{A} \mathbf{w}}{\mathbf{1}^T \mathbf{A} \mathbf{1}} \right), \quad \text{s.t. } \mathbf{1}^T \mathbf{A} \mathbf{w} = \mathbf{w}^T \mathbf{A} \mathbf{1} = 0, \tag{12}$$

where $\mathbf{w} = \{1, -b\}$. Introducing the diagonal matrices $\mathbf{D}_1 = diag(\mathbf{A}\mathbf{1})$ and $\mathbf{D}_2 = diag(\mathbf{1}^T \mathbf{A})$, we get a sum of the generalized Rayleigh quotients:

$$\hat{\mathbf{w}} = \min_{\mathbf{w}} \left(\frac{\mathbf{w}^T (\mathbf{D}_1 - \mathbf{A}) \mathbf{w}}{\mathbf{w}^T \mathbf{D}_1 \mathbf{w}} + \frac{\mathbf{w}^T (\mathbf{D}_2 - \mathbf{A}) \mathbf{w}}{\mathbf{w}^T \mathbf{D}_2 \mathbf{w}} \right) \tag{13}$$

that, in the standard eigenvector decomposition, applying the Cholesky decomposition to the matrices $\mathbf{D}_1$ and $\mathbf{D}_2$, is:

$$\left(\mathbf{D}_1^{-\frac{1}{2}} (\mathbf{D}_1 - \mathbf{A}) \mathbf{D}_1^{-\frac{1}{2}} + \mathbf{D}_2^{-\frac{1}{2}} (\mathbf{D}_2 - \mathbf{A}) \mathbf{D}_2^{-\frac{1}{2}} \right) \mathbf{w} = \lambda \mathbf{w} \tag{14}$$

In the case of a symmetric affinity, Eq. 14 provides the same solution for the vector $\mathbf{w}$ as Eq. 7. Figure 1(c) shows an example of correctly separated clusters obtained via asymmetric affinity in Figure 1(d).

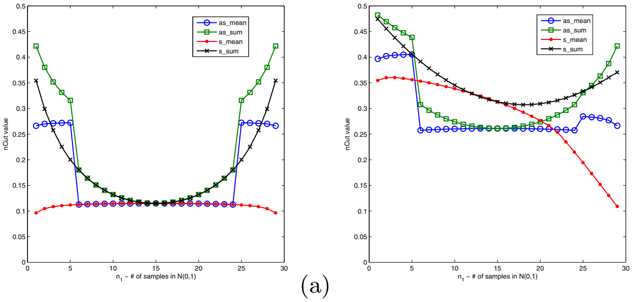
The defined cut can be utilized for hierarchical unsupervised learning, see Algorithm 1. Let a function $split(\mathbf{S}_i)$ bipartite the i^{th} subset of vertices $\mathbf{S}_i$, where $\mathbf{S}_i \subseteq \mathbf{S}$, into two disjoint subsets $\{\mathbf{S}_{10i+1}, \mathbf{S}_{10i+2}\}$ that satisfies their minimum cut value $nCut(\mathbf{S}_{10i+1}, \mathbf{S}_{10i+2})$. Applying data bipartition for each recovered subgraph, along with stopping conditions (minimum on the normalized cut or subgraph size), we can recursively split dataset $\mathbf{S}$ and present it in form of a binary tree of disjoint subgraphs. Usually, as a stopping criterion for clustering problems one tests the normalized graph cut value against some preassigned threshold, e.g. in [12] for image segmentation problems.

Algorithm 1. Hierarchical clustering for unlabeled data

```
S          // initial cluster contains complete dataset X
U = {1}    // set of cluster indices to process: S = {S_U} = {S_1}
V = ∅      // set of recovered cluster indices: V = ∅
while |U| ≠ ∅ do
    for k = 1 : |U| do
        [S_{10U_k+1}, S_{10U_k+2}] = split(S_{U_k})
        if nCut(S_{10U_k+1}, S_{10U_k+2}) ≤ threshold then
            U = U \ U_k
            U = U ∪ {10U_k + 1, 10U_k + 2}
        else
            V = V ∪ U_k
        end if
    end for
end while
```

Below we provide a numerical analysis of the normalized graph cut value for symmetric and asymmetric affinity matrices depending on the number of data samples in clusters for bipartition problem. Suppose we have n 1-dimensional data samples generated with two Gaussian distributions $N(m_1, \sigma_1)$, $N(m_2, \sigma_2)$, respectively. Here we study the normalized graph cuts depending on the number of samples in each cluster, such that $n = n_1 + n_2$, where n_1 and n_2 are the number of samples from each of the Gaussian distributions. From the properties of the normalized cut (3), we know that $nCut$ preserves small isolated subsets, whereas Cut tends to separate outliers. Generally speaking, $nCut$ is the maximum a posteriori estimation because its value depends on the number of entries in submatrices TL, TR, BL, BR, see Figure 1. To recover the maximum likelihood estimator from the $nCut$ we propose to modify Eq. 10, and instead of the *sum* operator we use the *mean* one for each block-matrix.

Figure 2(a) presents results for different cut values for $n = 30$, $N(0, 1)$, $N(5, 1)$, applied to symmetric and asymmetric affinity matrices. Indeed, the normalized cut based on the *sum* arises for marginal clusters, while $nCut$ based on *mean*



(a) (b)

Fig. 2. Normalized graph cut values for (a) symmetric and (b) asymmetric data distributions with varying samples ratio: (a) $N(0,1)$, $N(5,1)$; (b) $N(0,1)$, $N(5,3)$. Overall number of generated data samples from each cluster: $n_1 + n_2 = 30$. Symbols $\times$, $\bullet$, $\Box$, $\circ$ denote normalized graph cuts for symmetric sum, symmetric mean, asymmetric sum, and asymmetric mean affinity matrices, respectively

preserves constant cut value for all cluster sizes larger than $m = \sqrt{n}$. For this simple example, when clusters have identical but shifted distributions, we see that *nCut* based on the *mean* for asymmetric affinity matrix has almost constant values and does not depend on the size of the clusters when $n_1 > m$ $n_2 > m$. In the case of asymmetric distributions, see Figure 2(b), we generate data for $n = 30$, $N(0,1)$, $N(5,3)$. It is easy to see from these graphics that only the proposed *nCut* based on the *mean* for asymmetric affinity matrix does not depend on the size of recovered data clusters.

Algorithm 2 describes a recursive procedure for data ranking, e.g. EEG channel selection in BCI tasks, that aims to increase classification accuracy. Maximization of classification accuracy is equivalent to mutual information (MI) maximization of the class-labeled data [5]. MI monotonically increases with channel subset augmentation [5]. Knowing the class labels of the data, we can evaluate intra- and inter-class variance via an affinity matrix, so *nCut* is minimized only for the most informative (separated) data channels and has a unique global minimum. In the case of EEG recording, the proposed selection procedure is able to rank not only channels but also its samples, e.g. for EEG P300 responses it identifies both delay and length of the temporal window.

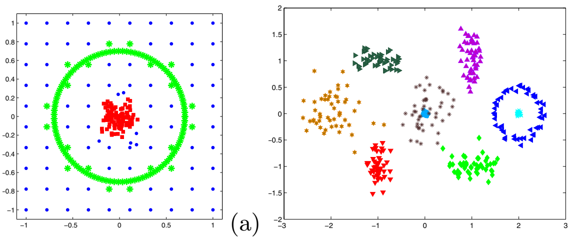
4 Numerical Results

To validate the proposed normalized cut we tested it on toy and real datasets for clustering and ranking problems[1]. Figure 3 shows clustering results based on the proposed method for overlapping and duplicated data distributions. The use of an asymmetric affinity matrix leads to good data separation.

[1] Matlab code, demo, and datasets are available on: `https://sites.google.com/site/kyrgyzov/cut`

Algorithm 2. Ranking procedure for class labeled data

$dir = 1$; // direction (1 - forward; -1 - backward)
n // the number of channels (or samples)
$\mathbf{X}$ // dataset consisting of n channels and $\mathbf{C} = \{0, 1\}$ classes
$\mathbf{U} = \{1, \ldots, n\}$; // initial set of channel indices
$\mathbf{V} = \emptyset$; // ranked set of channel indices
while $|\mathbf{U}| \neq \emptyset$ **do**
  for $k = 1 : |\mathbf{U}|$ **do**
    if $dir = 1$ **then**
      $nC(k) = nCut(\mathbf{X}[\mathbf{V} \cup \mathbf{U}_k | \mathbf{C}_1], \mathbf{X}[\mathbf{V} \cup \mathbf{U}_k | \mathbf{C}_2])$
    else
      $nC(k) = nCut(\mathbf{X}[\mathbf{U} \setminus \mathbf{U}_k | \mathbf{C}_1], \mathbf{X}[\mathbf{U} \setminus \mathbf{U}_k | \mathbf{C}_2])$
    end if
  end for
  $i = \min_i nC$ //index of a channel with minimal nCut
  $\mathbf{V} = \mathbf{V} \cup \mathbf{U}_i$
  $\mathbf{U} = \mathbf{U} \setminus \mathbf{U}_i$
end while



(a) (b)

Fig. 3. Data clustering via *nCut* based on asymmetric affinity matrix: (a) 3 clusters with overlapping; (b) 9 clusters with duplications and overlappings

Figure 4 shows image segmentations of the original 80×100 image from [12]. We preserve rules from [12] to evaluate the symmetric affinity matrix (multiplied symmetric affinities of brightness value of pixels and their spatial location) and perform partitioning with the same threshold value ($nCut \leq 0.04$). Subplot (d) shows the segmentation results based on asymmetric affinity (multiplied asymmetric affinity of brightness value of pixels and symmetric affinity of their spatial location), with limit on *nCut* based on the *mean* operator. Asymmetric affinity represents data diversity that allows us to identify image segments more efficiently.

Results for class-labeled EEG datasets show the usefulness of the proposed normalized graph cut for non-stationary signals corrupted by noise and artifacts [4]. In fact, we perform only data ranking and do not change classification methods. The following results, which are grounded on the informative EEG channels selection procedure, demonstrate its accuracy and efficiency. For the presented experiment, we used EEG P300 speller provided in [10] (winners of the BCI Competition III, Dataset II). It estimates the classification rate for an

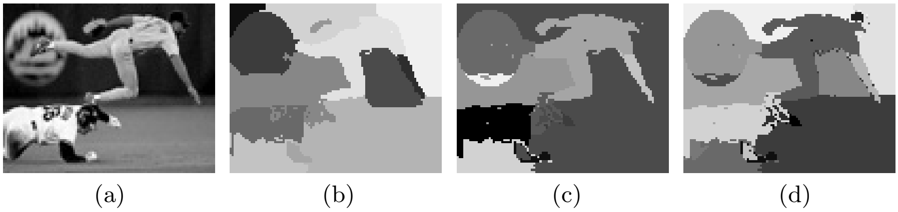


Fig. 4. Image segmentation via *nCut* with testing against predefined threshold based on: (b) symmetric affinity with limit on *nCut* based on *sum*; (c) symmetric affinity with limit on *nCut* based on *mean*; (d) asymmetric affinity with limit on *nCut* based on *mean*

ensemble of SVMs over 2 subjects with 64 EEG channels. Based on Algorithm 2, we perform data ranking for each of two subjects and utilize an increasing set of the most informative ordered channels to estimate their classification accuracy. Training datasets for this 2-class EEG classification problem are acquired with 64 channels and define 85 labeled cognitive states. After data preprocessing each channel for one P300 response consists of 14 samples. To estimate the normalized graph cut value for such dataset we calculate 64 affinity matrices for each of the EEG channels, so we have $[85 \times 85 \times 64]$ data cube $\mathbf{D}$. The final affinity $[85 \times 85]$ matrix $\mathbf{A}$ is the mean value over the 3^{rd} dimension (EEG channels) of the data cube $\mathbf{D}$. There are many ways to calculate the final affinity matrix for such datasets. In fact, such a 3D global affinity model allows us to rank not only EEG channels but also data samples in each of the channels, so we can identify the most informative observation window for P300 data records. Analyzing 3D affinity structure by its labeled dimensions, it is also possible to identify impostors in data record. Results are displayed in Fig 5.

The minima on the *nCut* graphs correspond to the optimal number of EEG channels. The classification accuracy graph dynamics are inverse to those of the normalized graph cut. This figure shows that a good accuracy can be achieved with a reduced number of channels: 20 and 15 for Subjects 1 and 2, respectively.

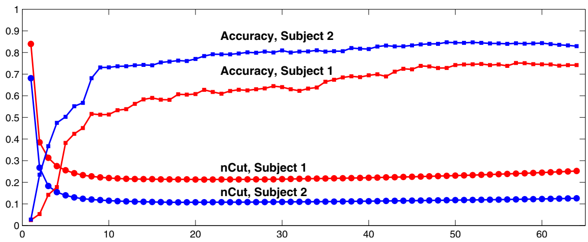


Fig. 5. Classification rate and *nCut* based on asymmetric affinity matrix vs. dimensionality of optimally selected channels

5 Conclusion

In this paper, we proposed a modification of the normalized graph cut and showed its usefulness to unsupervised learning as well as class-labeled data ranking tasks. Advantages of the modified approach with respect to unsupervised learning for overlapping and unequal data distributions were shown. We also provided a modified normalized cut estimation for asymmetric matrix based on the mean operator for hierarchical graph clustering. The algorithm for data ranking is a calibration step with small computing time for further classification. It selects the most informative parts of the data, which is critical for real-world problems, and does not require any prior information on classification environment, strategy and subjects.

References

1. Bollobas, B.: Modern graph theory (Graduate Texts in Mathematics), vol. 184. Springer (1998)
2. Forsyth, D., Ponce, J.: Computer Vision: A Modern Approach, 2nd edn. Prentice Hall (October 2011)
3. Golub, G., Loan, C.V.: Matrix Computations. John Hopkins Press (1989)
4. Kyrgyzov, O., Souloumiac, A.: Adaptive EEG artifact rejection for cognitive games. In: 14th ACM International Conference on Multimodal Interaction, ICMI 2012, pp. 567–570 (2012)
5. Lan, T., Erdogmus, D., Adami, A., Pavel, M., Mathan, S.: Salient EEG channel selection in brain computer interfaces by mutual information maximization. In: 27th Annual Conference on Engineering in Medicine and Biology, pp. 7064–7067 (2005)
6. Learned-Miller III, E., Fisher, J.: ICA using spacings estimates of entropy. Journal of Machine Learning Research 4, 1271–1295 (2003)
7. Meila, M., Pentney, W.: Clustering by weighted cuts in directed graphs. In: SIAM International Conference on Data Mining, pp. 135–144 (2007)
8. Nascimento, M., de Carvalho, A.: Spectral methods for graph clustering - a survey. European Journal of Operational Research 211(2), 221–231 (2011)
9. Ng, A., Jordan, M., Weiss, Y.: On spectral clustering: Analysis and an algorithm. In: Proc. 14th Advances in Neural Information Processing Systems, pp. 849–856 (2001)
10. Rakotomamonjy, A., Guigue, V.: BCI competition III: Dataset II - ensemble of SVMs for BCI P300 speller. IEEE Transactions on Biomedical Engineering 5(3), 1147–1154 (2008)
11. Schaeffer, S.: Graph clustering. Computer Science Review 1(1), 27–64 (2007)
12. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(8), 888–905 (2000)
13. Silverman, B.W.: Density Estimation for Statistics and Data Analysis. Chapman & Hall, London (1986)
14. Wu, Z., Leahy, R.: An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence 15(11), 1101–1113 (1993)
15. Yu, S., Shi, J.: Grouping with directed relationships. In: Figueiredo, M., Zerubia, J., Jain, A.K. (eds.) EMMCVPR 2001. LNCS, vol. 2134, pp. 283–297. Springer, Heidelberg (2001)