Spectral Analysis of Signed Graphs for Clustering, Prediction and Visualization

Jérôme Kunegis* Stephan Schmidt* Andreas Lommatzsch*
Jürgen Lerner† Ernesto W. De Luca* Sahin Albayrak*

Abstract

We study the application of spectral clustering, prediction and visualization methods to graphs with negatively weighted edges. We show that several characteristic matrices of graphs can be extended to graphs with positively and negatively weighted edges, giving signed spectral clustering methods, signed graph kernels and network visualization methods that apply to signed graphs. In particular, we review a signed variant of the graph Laplacian. We derive our results by considering random walks, graph clustering, graph drawing and electrical networks, showing that they all result in the same formalism for handling negatively weighted edges. We illustrate our methods using examples from social networks with negative edges and bipartite rating graphs.

1 Introduction

Many graph-theoretic data mining problems can be solved by spectral methods which consider matrices associated with a given network and compute their eigenvalues and eigenvectors. Common examples are spectral clustering, graph drawing and graph kernels used for link prediction and link weight prediction. In the usual setting, only graphs with unweighted or positively weighted edges are supported. Some data mining problems however apply to signed graphs, i.e. graphs that contain positively and negatively weighted edges. In this paper, we show that many spectral machine learning methods can be extended to the signed case using specially-defined characteristic matrices.

Intuitively, a positive edge in a graph denotes proximity or similarity, and a negative edge denotes dissimilarity or distance. Negative edges are found in many types of networks: social networks may contain not just "friend" but also "foe" links, and connections between users and products may denote like and dislike. In other cases, negative edges may be introduced explicitly, as in the case of constrained clustering, where some pairs of points *must* or *must not* be in the same cluster. These problems can all be modeled with signed graphs.

Spectral graph theory is the study of graphs using methods of linear algebra [4]. To study a given graph, its edge set is represented by an adjacency matrix, whose eigenvectors and eigenvalues are then used. Alternatively, the Laplacian matrix or one of several normalized adjacency matrices are used. In this paper, we are concerned with spectral graph mining algorithms that apply to signed graphs. In order to do this, we define variants of the Laplacian and related matrices that result in signed variants of spectral clustering, prediction and visualization methods.

We begin the study in Section 2 by considering the problem of drawing a graph with negative edges and derive the signed combinatorial graph Laplacian. In Section 3, we give the precise definitions of the various signed graph matrices, and derive basic results about their spectrum in Section 4. In Section 5 we derive the normalized and unnormalized signed Laplacian from the problem of signed spectral clustering using signed extensions of the ratio cut and normalized cut functions. In Section 6, we define several graph kernels that apply to signed graphs and show how they can be applied to link sign prediction in signed unipartite and bipartite networks. In Section 7 we give a derivation of the signed Laplacian using electrical networks in which resistances admit negative values. We conclude in Section 8.

2 Signed Graph Drawing

To motivate signed spectral graph theory, we consider the problem of drawing signed graphs, and show how it naturally leads to the signed normalized Laplacian. Proper definitions are given in the next section. The Laplacian matrix turns up in graph drawing when we try to find an embedding of a graph into a plane in a way that adjacent nodes are drawn near to each other [1]. The drawing of signed graphs is covered for instance in [2], but using the adjacency matrix instead of the Laplacian. In our approach, we stipulate that negative

*DAI-Labor, Technische Universität Berlin, Germany
†Universität Konstanz, Germany

edges should be drawn as far from each other as possible.

2.1 Positively Weighted Graphs We now describe the general method for generating an embedding of the nodes of a graph into the plane using the Laplacian matrix.

Given a connected graph $G = (V, E, A)$ with positively weighted edges, its adjacency matrix (A_{ij}) gives the positive edge weights when the vertices i and j are connected, and is zero otherwise. We now want to find a two-dimensional drawing of the graph in which each vertex is drawn near to its neighbors. This requirement gives rise to the following vertex equation, which states that every vertex is placed at the mean of its neighbors' coordinates, weighted by the weight of the connecting edges. For each node i, let $X_i \in \mathbb{R}^2$ be its coordinates in the drawing, then

$$X_i = \left(\sum_{j\sim i} A_{ij}\right)^{-1} \sum_{j\sim i} A_{ij} X_j. \tag{2.1}$$

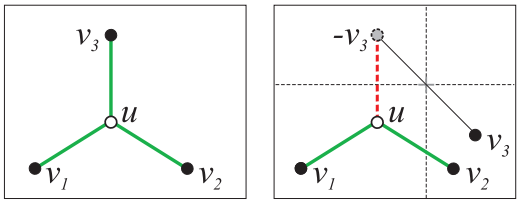
Rearranging and aggregating the equation for all i we arrive at

$$\begin{aligned} Dx &= Ax \\ Lx &= 0 \end{aligned} \tag{2.2}$$

where D is the diagonal matrix defined by $D_{ii} = \sum_j A_{ij}$ and $L = D - A$ is the combinatorial Laplacian of G. In other words, X should belong to the null space of L, which leads to the degenerate solution of X containing constant vectors, as the constant vector is an eigenvector of L with eigenvalue 0. To exclude that solution, we require additionally that the column vectors of X are orthogonal to the constant vector, leading to X being the eigenvectors associated with the two smallest eigenvalues of L different from zero. This solution results in a well-known satisfactory embedding of positively weighted graphs. Such an embedding is related to the resistance distance (or commute time distance) between nodes of the graph [1].

2.2 Signed Graphs We now extend the graph drawing method described in the previous section to graphs with positive and negative edge weights.

To adapt (2.1) for negative edge weights, we interpret a negative edge as an indication that two vertices should be placed on opposite sides of the drawing. Therefore, we take the opposite coordinates $-X_j$ of vertices j adjacent to i through a negative edge, and use the absolute value of edge weights to compute the mean, as pictured in Figure 1. We will call this construction



(a) Unsigned graph (b) Signed graph

Figure 1: Drawing a vertex at the mean coordinates of its neighbors by proximity and antipodal proximity. (a) In unsigned graphs, a vertex u is placed at the mean of its neighbors v_1, v_2, v_3. (b) In signed graphs, a vertex u is placed at the mean of its positive neighbors v_1, v_2 and antipodal points $-v_3$ of its negative neighbors.

antipodal proximity. This leads to the vertex equation

$$X_i = \left(\sum_{j\sim i} |A_{ij}|\right)^{-1} \sum_{j\sim i} A_{ij} X_j \tag{2.3}$$

resulting in a signed Laplacian matrix $\bar{L} = \bar{D} - A$ in which we take $\bar{D}_{ii} = \sum_j |A_{ij}|$:

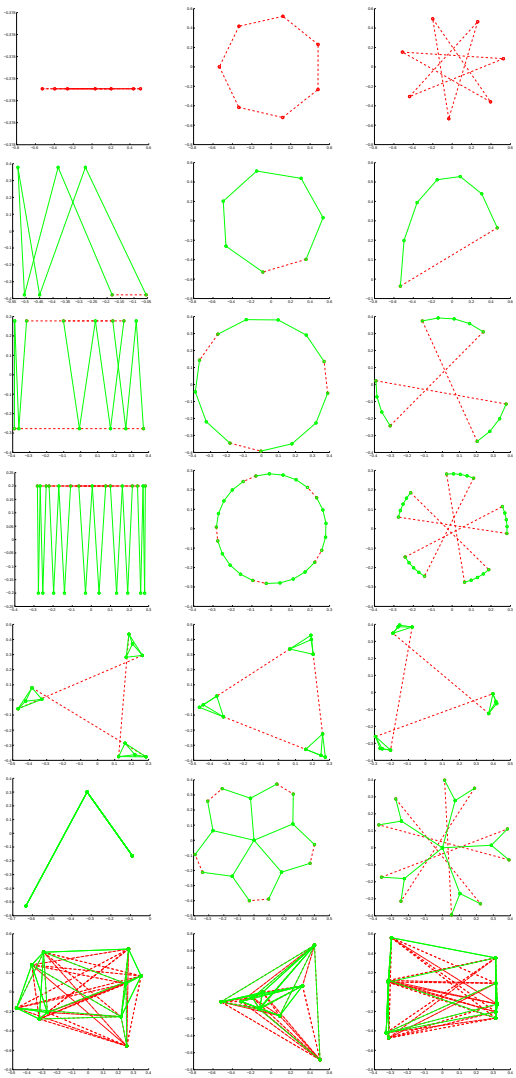
$$\begin{aligned} \bar{D}x &= Ax \\ \bar{L}x &= 0. \end{aligned} \tag{2.4}$$

As with L, we must now choose eigenvalues of $\bar{L}$ close to zero as coordinates. As we will see in the next section, $\bar{L}$ is always positive-semidefinite, and is positive-definite for graphs that are unbalanced, i.e. that contain cycles with an odd number of negative edges.

To obtain a graph drawing from $\bar{L}$, we can thus distinguish three cases, assuming that G is connected:

- If all edges are positive, $\bar{L} = L$ and we arrive at the solution given by the ordinary Laplacian matrix.
- If the graph is unbalanced, $\bar{L}$ is positive-definite and we can use the eigenvectors of the two smallest eigenvalues as coordinates.
- If the graph is balanced, its spectrum is equivalent to that of the corresponding unsigned Laplacian matrix, up to signs of the eigenvector components. Using the eigenvectors of the two smallest eigenvalues (including zero), we arrive at a graph drawing with all points being placed on two parallel lines, reflecting the perfect 2-clustering present in the graph.

2.3 Toy Examples Figure 2 shows example graphs with positive edges drawn in green and negative edges in red. All edges have weight ± 1, and all graphs contain cycles with an odd number of negative edges. Column (a) shows all graphs drawn using the eigenvectors of the two largest eigenvalues of the adjacency matrix A. Column (b) shows the unsigned Laplacian embedding of the graphs by setting all edge weights to $+1$. Column (c) shows the signed Laplacian embedding. The embedding given by the eigenvectors of A is clearly not satisfactory for graph drawing. As expected, the graphs drawn using the ordinary Laplacian matrix place nodes connected by a negative edge near to each other. The signed Laplacian matrix produces a graph embedding where negative links span large distances across the drawing, as required.



(a) A (b) L (c) $\bar{L}$

Figure 2: Small toy graphs drawn using the eigenvectors of three graph matrices. (a) The adjacency matrix A. (b) The unsigned Laplacian $D - A$. (c) The signed Laplacian $\bar{D} - A$. All graphs shown contain negative cycles, and their signed Laplacian matrices are positive-definite. Edges have weights ± 1 shown in green $(+1)$ and red (-1).

3 Definitions

In this section, we give the definition of the combinatorial and normalized signed Laplacian matrices of a graph and derive their basic properties.

The combinatorial Laplacian matrix of signed graphs with edge weights restricted to ± 1 is described in [12] where it is called the Kirchhoff matrix (of a signed graph). A different Laplacian matrix for signed graphs that is not positive-semidefinite is used in the context of knot theory [19]. That same Laplacian matrix is used in [16] to draw graphs with negative edge weights. For graphs where all edges have negative weights, the matrix $D + A$ may be considered, and corresponds to the Laplacian of the underlying unsigned graph [5].

Let $G = (V, E, A)$ be an undirected graph with vertex set V, edge set E, and nonzero edge weights described by the adjacency matrix $A \in \mathbb{R}^{V \times V}$. We will denote an edge between nodes i and j as (i, j) and write $i \sim j$ to signify that two nodes are adjacent. If (i, j) is not an edge of the graph, we set $A_{ij} = 0$. Otherwise, $A_{ij} > 0$ denotes a positive edge and $A_{ij} < 0$ denotes a negative edge. Unless otherwise noted, we assume G to be connected.

3.1 Laplacian Matrix Given a graph G with only positively weighted edges, its ordinary Laplacian matrix is a symmetric $V \times V$ matrix that, in a general sense, captures relations between individual nodes of the graph. The Laplacian matrix is positive-semidefinite, and its Moore–Penrose pseudoinverse can be interpreted as a forest count [3] and can be used to compute the resistance distance between any two nodes [14].

DEFINITION 1. *The Laplacian matrix $L \in \mathbb{R}^{V \times V}$ of a graph G with nonnegative adjacency matrix A is given by*

$$L = D - A \tag{3.5}$$

with the diagonal degree matrix $D \in \mathbb{R}^{V \times V}$ given by

$$D_{ii} = \sum_{j \sim i} A_{ij}. \tag{3.6}$$

The multiplicity of the Laplacian's eigenvalue zero equals the number of connected components in G. If G

is connected, the eigenvalue zero has multiplicity one, and the second-smallest eigenvalue is known as the algebraic connectivity of the graph [4]. The eigenvector corresponding to that second-smallest eigenvalue is called the *Fiedler vector*, and has been used successfully for clustering the nodes of G [6].

3.2 Signed Laplacian Matrix If applied to signed graphs, the Laplacian of Equation (3.5) results in an indefinite matrix, and thus cannot be used as the basis for graph kernels [19]. Therefore, we use a modified degree matrix $\bar{D}$ [12].

DEFINITION 2. *The signed Laplacian matrix $\bar{L} \in \mathbb{R}^{V \times V}$ of a graph G with adjacency matrix A is given by*

$$\bar{L} = \bar{D} - A, \tag{3.7}$$

where the signed degree matrix $\bar{D} \in \mathbb{R}^{V \times V}$ is the diagonal matrix given by

$$\bar{D}_{ii} = \sum_{j \sim i} |A_{ij}|. \tag{3.8}$$

We prove in Section 4 that the signed Laplacian matrix is positive-semidefinite.

Two different matrices are usually called the normalized Laplacian, and both can be extended to signed graphs. We follow [20] and call these the random walk and symmetric normalized Laplacian matrices.

3.3 Random Walk Normalized Laplacian When modeling random walks on an unsigned graph G, the transition probability from node i to j is given by entries of the stochastic matrix $D^{-1}A$. This matrix also arises from Equation (2.2) as the eigenvalue equation $x = D^{-1}Ax$. The matrix $L_{\text{rw}} = I - D^{-1}A$ is called the random walk Laplacian and is positive-semidefinite. Its signed counterpart is given by $\bar{L}_{\text{rw}} = I - \bar{D}^{-1}A$.

The random walk normalized Laplacian arises when considering random walks, but also when drawing graphs and when clustering using normalized cuts, as explained in Section 5.

3.4 Symmetric Normalized Laplacian Another signed Laplacian is given by $L_{\text{sym}} = I - D^{-1/2}AD^{-1/2}$. As with the unnormalized Laplacian matrix, we define an ordinary and a signed variant.

$$\begin{aligned} L_{\text{sym}} &= D^{-1/2}LD^{-1/2} = I - D^{-1/2}AD^{-1/2} \\ \bar{L}_{\text{sym}} &= \bar{D}^{-1/2}\bar{L}\bar{D}^{-1/2} = I - \bar{D}^{-1/2}A\bar{D}^{-1/2} \end{aligned}$$

The ordinary variant only applies to graphs with positive edge weights. The normalized Laplacian matrices can be used instead of the combinatorial Laplacian matrices in most settings, with good results reported for graphs with very skewed degree distributions [8].

4 Spectral Analysis

In this section, we prove that the signed Laplacian matrix L is positive-semidefinite, characterize the graphs for which it is positive-definite, and give the relationship between the eigenvalue decomposition of the signed Laplacian matrix and the eigenvalue decomposition of the corresponding unsigned Laplacian matrix. The characterization of the smallest eigenvalue of $\bar{L}$ in terms of graph balance can be found in [12].

4.1 Positive-semidefiniteness

THEOREM 4.1. *The signed Laplacian matrix $\bar{L}$ is positive-semidefinite for any graph G.*

Proof. We write the Laplacian matrix as a sum over the edges of G:

$$\bar{L} = \sum_{(i,j) \in E} \bar{L}^{(i,j)}$$

where $\bar{L}^{(i,j)} \in \mathbb{R}^{V \times V}$ contains the four following nonzero entries:

$$\begin{aligned} \bar{L}^{(i,j)}_{ii} = \bar{L}^{(i,j)}_{jj} &= |A_{ij}| \\ \bar{L}^{(i,j)}_{ij} = \bar{L}^{(i,j)}_{ji} &= -A_{ij}. \end{aligned}$$

Let $x \in \mathbb{R}^V$ be a vertex-vector. By considering the bilinear form $x^T\bar{L}^{(i,j)}x$, we see that $\bar{L}^{(i,j)}$ is positive-semidefinite:

$$\begin{aligned} x^T\bar{L}^{(i,j)}x &= |A_{ij}|x_i^2 + |A_{ij}|x_j^2 - 2A_{ij}x_ix_j \\ &= |A_{ij}|(x_i - \text{sgn}(A_{ij})x_j)^2 \\ &\geq 0 \end{aligned} \tag{4.9}$$

We now consider the bilinear form $x^T\bar{L}x$:

$$x^T\bar{L}x = \sum_{(i,j) \in E} x^T\bar{L}^{(i,j)}x \geq 0$$

It follows that $\bar{L}$ is positive-semidefinite.

Another way to prove that $\bar{L}$ is positive-semidefinite consists of expressing it using the incidence matrix of G. Assume that for each edge (i,j), an arbitrary orientation is chosen. Then we define the incidence matrix $S \in \mathbb{R}^{V \times E}$ of G as

$$\begin{aligned} S_{i(i,j)} &= +\sqrt{|A_{ij}|} \\ S_{j(i,j)} &= -\text{sgn}(A_{ij})\sqrt{|A_{ij}|}. \end{aligned}$$

We now consider the product $SS^T \in \mathbb{R}^{V \times V}$:

$$\begin{aligned} (SS^T)_{ii} &= \sum_{j \sim i} |A_{ij}| \\ (SS^T)_{ij} &= -A_{ij} \end{aligned}$$

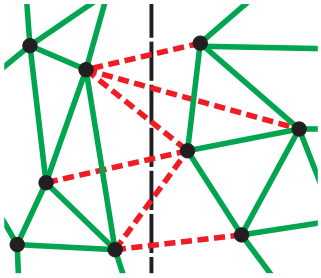


Figure 3: The nodes of a graph without negative cycles can be partitioned into two sets such that all edges inside of each group are positive and all edges between the two groups are negative. We call such a graph balanced, and the eigenvalue decomposition of its signed Laplacian matrix can be expressed as the modified eigenvalue decomposition of the corresponding unsigned graph's Laplacian.

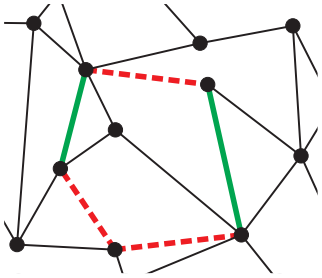


Figure 4: An unbalanced graph contains a cycle with an odd number of negatively weighted edges. Negatively weighted edges are shown in red, positively weighted edges are shown in green, and edges that are not part of the cycle in black. The presence of such cycles results in a positive-definite Laplacian matrix.

for diagonal and off-diagonal entries, respectively. Therefore $SS^T = \bar{L}$, and it follows that $\bar{L}$ is positive-semidefinite. This result is independent of the orientation chosen for S.

4.2 Positive-definiteness We now show that, unlike the ordinary Laplacian matrix, the signed Laplacian matrix is strictly positive-definite for some graphs, including most real-world networks.

As with the ordinary Laplacian matrix, the spectrum of the signed Laplacian matrix of a disconnected graph is the union of the spectra of its connected components. This can be seen by noting that the Laplacian matrix of an unconnected graph has block-diagonal form, with each diagonal entry being the Laplacian matrix of a single component. Therefore, we will restrict ourselves to connected graphs. First, we define balanced graphs [11].

DEFINITION 3. *A connected graph with nonzero signed edge weights is balanced when its vertices can be partitioned into two groups such that all positive edges connect vertices within the same group, and all negative edges connect vertices of different groups.*

Figure 3 shows a balanced graph partitioned into two vertex sets. Equivalently, unbalanced graphs can be defined as those graphs containing a cycle with an odd number of negative edges, as shown in Figure 4.

To prove that the balanced graphs are exactly those that do not contain cycles with an odd number of edges, consider that any cycle in a balanced graph has to cross sides an even number of times. On the other hand, any balanced graph can be partitioned into two vertex sets by depth-first traversal while assigning each vertex to a partition such that the balance property is fulfilled. Any inconsistency that arises during such a labeling leads to a cycle with an odd number of negative edges.

Using this definition, we can characterize the graphs for which the signed Laplacian matrix is positive-definite.

THEOREM 4.2. *The signed Laplacian matrix of an unbalanced graph is positive-definite.*

Proof. We show that if the bilinear form $x^T \bar{L} x$ is zero for some $x \neq 0$, then a bipartition of the vertices as described above exists.

Let $x^T \bar{L} x = 0$. We have seen that for every $\bar{L}^{(i,j)}$ and any x, $x^T \bar{L}^{(i,j)} x \geq 0$. Therefore, we have for every edge (i, j)

$$\begin{aligned} x^T \bar{L}^{(i,j)} x &= 0 \\ |A_{ij}|(x_i - \operatorname{sgn}(A_{ij}) x_j)^2 &= 0 \\ x_i &= \operatorname{sgn}(A_{ij}) x_j \end{aligned}$$

In other words, two components of x are equal if the corresponding vertices are connected by a positive edge, and opposite to each other if the corresponding vertices are connected by a negative edge. Because the graph is connected, it follows that all $|x_i|$ must be equal. We can exclude the solution $x_i = 0$ for all i because x is not the zero vector. Without loss of generality, we assume that $|x_i| = 1$ for all i. Therefore, x gives a bipartition into vertices with $x_i = +1$ and vertices with $x_i = -1$, with the property that two vertices with the same value of x_i are in the same partition and two vertices with opposite sign of x_i are in different partitions, and therefore G is balanced.

4.3 Balanced Graphs We now show how the spectrum and eigenvalues of the signed Laplacian of a balanced graph arise from the spectrum and the eigenvalues of the corresponding unsigned graph by multiplication of eigenvector components with ± 1.

Let $G = (V, E, A)$ be a balanced graph with positive and negative edge weights and $\mathfrak{G} = (V, E, \mathfrak{A})$ the corresponding graph with positive edge weights given by $\mathfrak{A}_{ij} = |A_{ij}|$. Since G is balanced, there is a vector $x \in \{-1, +1\}^V$ such that for all edges (i, j), $\operatorname{sgn}(A_{ij}) = x_i x_j$.

THEOREM 4.3. *If $\bar{L}$ is the signed Laplacian matrix of the balanced graph G with bipartition x and eigenvalue decomposition $\bar{L} = U\Lambda U^T$, then the eigenvalue decomposition of the Laplacian matrix $\mathfrak{L}$ of $\mathfrak{G}$ of the corresponding unsigned graph $\mathfrak{G}$ of G is given by $\mathfrak{L} = \mathfrak{U}\Lambda\mathfrak{U}^T$ where*

$$\mathfrak{U}_{ij} = x_i U_{ij}. \tag{4.10}$$

Proof. To see that $\mathfrak{L} = \mathfrak{U}\Lambda\mathfrak{U}^T$, note that for diagonal elements, $\mathfrak{U}_i\Lambda\mathfrak{U}_i^T = x_i^2 U_i \Lambda U_i^T = U_i \Lambda U_i^T = \bar{L}_{ii} = \mathfrak{L}_{ii}$. For off-diagonal elements, we have $\mathfrak{U}_i\Lambda\mathfrak{U}_j^T = x_i x_j U_i \Lambda U_j^T = \operatorname{sgn}(A_{ij})\bar{L}_{ij} = -\operatorname{sgn}(A_{ij})A_{ij} = -|A_{ij}| = -\mathfrak{A}_{ij} = \mathfrak{L}_{ij}$.

We now show that $\mathfrak{U}\Lambda\mathfrak{U}^T$ is an eigenvalue decomposition of $\mathfrak{L}$ by showing that $\mathfrak{U}$ is orthogonal. To see that the columns of $\mathfrak{U}$ are indeed orthogonal, note that for any two column indices $m \neq n$, we have $\mathfrak{U}_{\cdot m}^T \mathfrak{U}_{\cdot n} = \sum_{i \in V} \mathfrak{U}_{im}\mathfrak{U}_{in} = \sum_{i \in V} x_i^2 U_{im} U_{in} = U_{\cdot m}^T U_{\cdot n} = 0$ because U is orthogonal. Changing signs in U does not change the norm of each column vector, and thus $\mathfrak{L} = \mathfrak{U}\Lambda\mathfrak{U}^T$ is a proper eigenvalue decomposition.

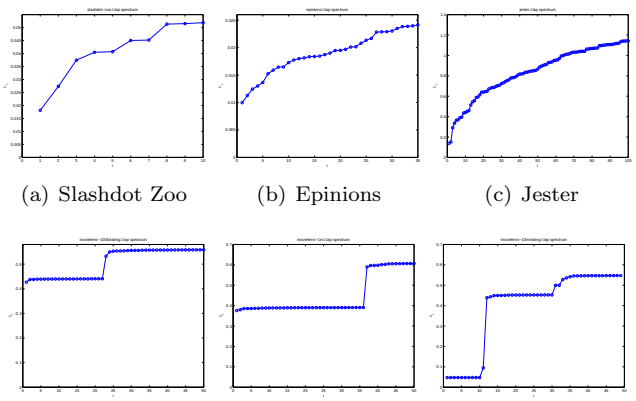
As shown in Section 4.2, the Laplacian matrix of an unbalanced graph is positive-definite and therefore its spectrum is different from that of the corresponding unsigned graph. Aggregating Theorems 4.2 and 4.3, we arrive at our main result.

THEOREM 4.4. *The signed Laplacian matrix of a graph is positive-definite if and only if the graph is unbalanced.*

Proof. The theorem follows directly from Theorems 4.2 and 4.3.

The spectra of several large unipartite and bipartite signed networks are plotted in Figure 5. Table 1 gives the smallest Laplacian eigenvalues for some large networks. Jester is a bipartite user–item network of joke ratings [9]. MovieLens is the bipartite user–item network of signed movie ratings from GroupLens research[1], released in three different sizes. The Slashdot Zoo is a social network with "friend" and "foe" links extracted from technology news website Slashdot [17]. Epinions is a signed user–user opinion network [21]. The smallest Laplacian eigenvalue is a characteristic value of the network denoting its balance. The higher the value the more unbalanced is the network [12]. All these large networks are unbalanced, and we can for instance observe that the social networks are more balanced than the rating networks.

[1] http://www.grouplens.org/node/73



(a) Slashdot Zoo (b) Epinions (c) Jester

(d) MovieLens 100k (e) MovieLens 1M (f) MovieLens 10M

Figure 5: The Laplacian spectra of several large signed networks.

Table 1: Smallest Laplacian eigenvalue $\lambda_1 > 0$ of large signed networks. Smaller values indicate a more balanced network.

Network	λ_1	
MovieLens 100k	0.4285	conflict
MovieLens 1M	0.3761	↑
Jester	0.06515	↑
MovieLens 10M	0.04735	↑
Slashdot Zoo	0.006183	↓
Epinions	0.004438	balance

5 Clustering

One of the main application areas of the graph Laplacian are clustering problems. In spectral clustering, the eigenvectors of matrices associated with a graph are used to partition the vertices of the graph into well-connected groups. In this section, we show that in a graph with negatively weighted edges, spectral clustering algorithms correspond to finding clusters of vertices connected by positive edges, but not connected by negative edges.

Spectral clustering algorithms are usually derived by formulating a minimum cut problem which is then relaxed. The choice of the cut function results in

different spectral clustering algorithms. In all cases, the vertices of a given graph are mapped into the space spanned by the eigenvector components of a matrix associated with the graph.

We derive a signed extension of the ratio cut, which leads to clustering with the signed combinatorial Laplacian $\bar{L}$. Analogous derivations are possible for the normalized Laplacians. We restrict our proofs to the case of 2-clustering. Higher-order clusterings can be derived analogously.

5.1 Unsigned Graphs We first review the derivation of the ratio cut in unsigned graphs, which leads to a clustering based on the eigenvectors of L.

Let $G = (V, E, A)$ be an unsigned graph. A cut of G is a partition of the vertices $V = X \cup Y$. The weight of a cut is given by $\text{cut}(X, Y) = \sum_{i \in X, j \in Y} A_{ij}$. The cut measures how well two clusters are connected. Since we want to find two distinct groups of vertices, the cut must be minimized. Minimizing $\text{cut}(X, Y)$ however leads in most cases to solutions separating very few vertices from the rest of the graph. Therefore, the cut is usually divided by the size of the clusters, giving the ratio cut:

$$\text{RatioCut}(X, Y) = \text{cut}(X, Y) \left(\frac{1}{|X|} + \frac{1}{|Y|} \right)$$

To get a clustering, we then solve the following optimization problem:

$$\min_{X \subset V} \text{RatioCut}(X, V \setminus X)$$

Let $Y = V \setminus X$, then this problem can be solved by expressing it in terms of the characteristic vector u of X defined by:

$$u_i = +\sqrt{|Y|/|X|} \text{ when } i \in X \qquad (5.11)$$
$$u_i = -\sqrt{|X|/|Y|} \text{ when } i \in Y$$

We observe that $uLu^T = 2|V| \cdot \text{RatioCut}(X, Y)$, and that $\sum_i u_i = 0$, i.e. u is orthogonal to the constant vector. Denoting by $\mathcal{U}$ the vectors u of the form given in Equation (5.11) we have

$$\min_{u \in \mathbb{R}^V} \quad uLu^T$$
$$\text{s.t.} \quad u \perp \mathbf{1}, u \in \mathcal{U}$$

This can be relaxed by removing the constraint $u \in \mathcal{U}$, giving as solution the eigenvector of L having the smallest nonzero eigenvalue [20]. The next subsection gives an analogous derivation for signed graphs.

5.2 Signed Graphs We now give a derivation of the ratio cut for signed graphs. Let $G = (V, E, A)$ be a signed graph. We will write A^+ and A^- for the adjacency matrices containing only positive and negative edges. In other words, $A^+_{ij} = \max(0, A_{ij})$ and $A^-_{ij} = \max(0, -A_{ij})$.

For convenience we define positive and negative cuts that only count positive and negative edges respectively:

$$\text{cut}^+(X, Y) = \sum_{i \in X, j \in Y} A^+_{ij}$$
$$\text{cut}^-(X, Y) = \sum_{i \in X, j \in Y} A^-_{ij}$$

In these definitions, we allow X and Y to be overlapping. For a vector $u \in \mathbb{R}^V$, we consider the bilinear form $u^T \bar{L} u$. As shown in Equation (4.9), this can be written in the following way:

$$u^T \bar{L} u = \sum_{i \sim j} |A_{ij}| (u_i - \text{sgn}(A_{ij}) u_j)^2$$

For a given partition (X, Y), let $u \in \mathbb{R}^V$ be the following vector:

$$u_i = +\frac{1}{2} \left(\sqrt{\frac{|A|}{|\bar{A}|}} + \sqrt{\frac{|\bar{A}|}{|A|}} \right) \text{ when } i \in X \qquad (5.12)$$
$$u_i = -\frac{1}{2} \left(\sqrt{\frac{|A|}{|\bar{A}|}} + \sqrt{\frac{|\bar{A}|}{|A|}} \right) \text{ otherwise.}$$

The corresponding bilinear form then becomes:

$$\begin{aligned} u^T \bar{L} u &= \sum_{i \sim j} |A_{ij}| \left(u_i - \text{sgn}(A_{ij}) u_j \right)^2 \\ &= |V| \left(2 \cdot \text{cut}^+(X, Y) + \text{cut}^-(X, X) + \text{cut}^-(Y, Y) \right) \\ &\quad \left(\frac{1}{|X|} + \frac{1}{|Y|} \right) \end{aligned}$$

This leads us to define the following signed cut of (X, Y):

$$\begin{aligned} \text{scut}(X, Y) &= 2 \cdot \text{cut}^+(X, Y) \\ &\quad + \text{cut}^-(X, X) + \text{cut}^-(Y, Y) \end{aligned}$$

and to define the signed ratio cut as follows:

$$\text{SignedRatioCut}(X, Y) = \text{scut}(X, Y) \left(\frac{1}{|X|} + \frac{1}{|Y|} \right)$$

Therefore, the following minimization problem solves the signed clustering problem, where $\mathcal{U}$ denotes vectors of the form given in Equation (5.12).

$$\min_{X \subset V} \quad \text{SignedRatioCut}(X, V \setminus X)$$
$$\min_{u \in \mathbb{R}^V} \quad u \bar{L} u^T \text{ s.t. } u \in \mathcal{U}$$

Note that we lose the orthogonality of u to the constant vector. This can be explained by the fact that when G contains negative edges, the smallest eigenvector can always be used for clustering: If G is balanced, the smallest eigenvalue is zero and its eigenvector equals (± 1) and gives the two clusters separated by negative edges. If G is unbalanced, the smallest eigenvalue is larger than zero, so the constant vector plays no role.

The signed cut $\mathrm{scut}(X, Y)$ counts the number of positive edges that connect the two groups X and Y, and the number of negative edges that remain in each of these groups. Thus, minimizing the signed cut leads to clusterings where two groups are connected by few positive edges and contain few negative edges inside each group. This signed ratio cut generalizes the ratio cut of unsigned graphs and justifies the use of the signed Laplacian $\bar{L}$ for spectral clustering of signed graphs.

5.3 Normalized Laplacian When instead of normalizing with the number of vertices $|X|$ we normalize with the number of edges $\mathrm{vol}(X)$, the result is a spectral clustering algorithm based on the eigenvectors of $D^{-1}A$ introduced by Shi and Malik [24]. The cuts normalized by $\mathrm{vol}(X)$ are called normalized cuts. In the signed case, the eigenvectors of $\bar{D}^{-1}A$ lead to the signed normalized cut:

$$\begin{aligned}&\mathrm{SignedNormalizedCut}(X, Y)\\ &= \mathrm{scut}(X, Y)\left(\frac{1}{\mathrm{vol}(X)} + \frac{1}{\mathrm{vol}(Y)}\right)\end{aligned}$$

A similar derivation can be made for normalized cuts based on $\bar{D}^{-1/2}A\bar{D}^{-1/2}$, generalizing the spectral clustering method of Ng, Jordan and Weiss [22]. The following section gives an example of clustering a small, signed graph.

5.4 Anthropological Example As an application of signed spectral clustering to real-world data, we present the dataset of [23]. This dataset describes the relations between sixteen tribal groups of the Eastern Central Highlands of New Guinea [10]. Relations between tribal groups in the Gahuku–Gama alliance structure can be friendly ("rova") or antagonistic ("hina"). We model the dataset as a graph with edge weights $+1$ for friendship and -1 for enmity.

The resulting graph contains cycles with an odd number of negative edges, and therefore its signed Laplacian matrix is positive-definite. We use the eigenvectors of the two smallest eigenvalues (1.04 and 2.10) to embed the graph into the plane. The result is shown in Figure 6. We observe that indeed the positive (green) edges are short, and the negative (red) edges are long. Looking at only the positive edges, the drawing makes the two connected components easy to see. Looking at only the negative edges, we recognize that the tribal groups can be clustered into three groups, with no negative edges inside any group. These three groups correspond indeed to a higher-order grouping in the Gahuku–Gama society [10]. An example on a larger network is shown in the next section, using the genre of movies in a user–item graph with positive and negative ratings.

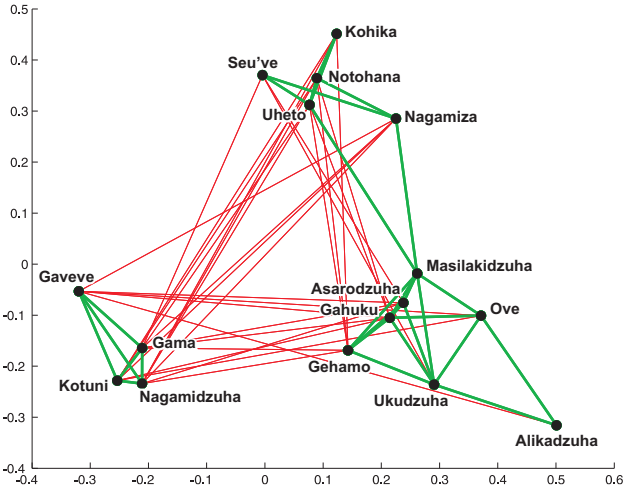


Figure 6: The tribal groups of the Eastern Central Highlands of New Guinea from the study of Read [23] drawn using signed Laplacian graph embedding. Individual tribes are shown as vertices of the graphs, with friendly relations shown as green edges and antagonistic relations shown as red edges. The three higher-order groups as described by Hage & Harary in [10] are linearly separable.

6 Graph Kernels and Link Prediction

In this section, we describe Laplacian graph kernels that apply to graphs with negatively weighted edges. We first review ordinary Laplacian graph kernels, and then extend these to signed graphs.

A kernel is a function of two variables $k(x, y)$ that is symmetric, positive-semidefinite, and represents a measure of similarity in the space of its arguments. Graph kernels are defined on the vertices of a given graph, and are used for link prediction, classification and other machine learning problems.

6.1 Unsigned Laplacian Kernels We briefly review three graph kernels based on the ordinary Laplacian matrix.

Commute time kernel The simplest graph kernel possible using L is the Moore–Penrose pseudoinverse L^+. This kernel is called the commute time kernel due

to its connection to random walks [7]. Alternatively, it can be described as the resistance distance kernel, based on interpreting the graph as a network of electrical resistances [14].

Regularized Laplacian kernel Regularizing the Laplacian kernel results in the regularized Laplacian graph kernel $(I + \alpha L)^{-1}$, which is always positive-definite [13]. α is the regularization parameter.

Heat diffusion kernel By considering a process of heat diffusion on a graph one arrives at the heat diffusion kernel $\exp(-\alpha L)$ [15].

6.2 Signed Laplacian Kernels To apply the Laplacian graph kernels to signed graphs, we replace the ordinary Laplacian L by the signed Laplacian $\bar{L}$.

Signed resistance distance In graphs with negative edge weights, the commute time kernel can be extended to $\bar{L}^+$. As noted in Section 4, $\bar{L}$ is positive-definite for certain graphs, in which case the pseudoinverse reduces to the ordinary matrix inverse. The signed Laplacian graph kernel can also be interpreted as the signed resistance distance kernel [18]. A separate derivation of the signed resistance distance kernel is given in the next section.

Regularized Laplacian kernel For signed graphs, we define the regularized Laplacian kernel as $(I + \alpha\bar{L})^{-1}$.

Heat diffusion kernel We extend the heat diffusion kernel to signed graphs giving $\exp(-\alpha\bar{L})$.

Because $\bar{L}$ is positive-semidefinite, it follows that all three kernels are also positive-semidefinite and indeed proper kernels.

In the following two subsections, we perform experiments in which we apply these signed Laplacian graph kernels to link sign prediction in unipartite and bipartite networks, respectively. Since these networks contain negatively weighted edges, we cannot apply the ordinary Laplacian graph kernels to them. Therefore, we compare the signed Laplacian graph kernels to other kernels defined for networks with negative edges.

6.3 Social Network Analysis In this section, we apply the signed Laplacian kernels to the task of link sign prediction in the social network of user relationships in the Slashdot Zoo. The Slashdot Zoo consists of the relationships between users of the technology news site Slashdot. On Slashdot, users can tag other users as "friends" and "foes", giving rise to a network of users connected by two edge types [17]. We model *friend* links as having the weight $+1$ and *foe* links as having the weight -1. We ignore link direction, resulting in an undirected graph. The resulting network has 71,523 nodes and 488,440 edges, of which 24.1% are negative.

6.3.1 Evaluation Setup As the task for evaluation, we choose the prediction of link sign. We split the edge set into a training and a test set, the test set containing 33% of all edges. In contrast to the usual task of link prediction, we do not try to predict whether a node is present in the test set, but only its weight.

As baseline kernels, we take the symmetric adjacency matrix A of the training set, compute its reduced eigenvalue decomposition $A_{(k)} = U\Lambda U^T$ of rank $k = 9$, and use it to compute the following three link sign prediction functions:

Rank reduction The reduced eigenvalue decomposition $A_{(k)}$ itself can be used to compute a rank-k approximation of A. We use this approximation as the link sign prediction.

Power sum We use a polynomial of degree 4 of $A_{(k)}$, the coefficients of which are determined empirically to give the most accurate link sign prediction by cross validation.

Matrix exponential We compute the exponential graph kernel $\exp(A_{(k)}) = U\exp(\Lambda)U^T$, and use it for link prediction.

To evaluate the signed spectral approach, we use three graph kernels based on the signed Laplacian matrix $\bar{L}$. These kernels are computed with the reduced eigenvalue decomposition of $\bar{L}_{(k)} = U\Lambda U^T$ in which only the smallest $k = 9$ eigenvalues are kept.

Signed resistance distance The pseudoinverse of the signed Laplacian $\bar{L}^+ = U\Lambda^+U^T$ gives the signed resistance distance kernel. In this dataset, odd cycles exist, and so $\bar{L}$ is positive-definite. We thus use all eigenvectors of $\bar{L}$.

Signed regularized Laplacian By regularization we arrive at $(I + \alpha\bar{L})^{-1} = U(I + \alpha\Lambda)^{-1}U^T$, the regularized Laplacian graph kernel.

Signed heat diffusion We compute the heat diffusion kernel given by $\exp(-\alpha\bar{L}) = U\exp(-\alpha\Lambda)U^T$.

6.3.2 Evaluation Results As a measure of prediction accuracy, we use the root mean squared error (RMSE) [7]. The evaluation results are summarized in Table 7. We observe that all three signed Laplacian graph kernels perform better than the baseline methods. The best performance is achieved by the signed regularized Laplacian kernel.

6.4 Collaborative Filtering To evaluate the signed Laplacian graph kernel on a bipartite network, we chose the task of collaborative filtering on the MovieLens 100k corpus, as published by GroupLens Research. In a collaborative filtering setting, the network consists of users, items and ratings. Each rating is represented by an edge between a user and an item. The MovieLens 100k cor-

Figure 7: The link sign prediction accuracy of different graph kernels measured by the root mean squared error (RMSE).

Kernel	RMSE
Rank reduction	0.838
Power sum	0.840
Matrix exponential	0.839
Signed resistance distance	0.812
Signed regularized Laplacian	0.778
Signed heat diffusion	0.789

pus of movie ratings contains 6.040 users, 3.706 items, and about a hundred thousand ratings.

Figure 8 shows the users and items of the MovieLens 100k corpus drawn with the method described in 2.2. We observe that the signed Laplacian graph drawing methods place movies on lines passing through the origin. These lines correspond to clusters in the underlying graph, as illustrated by the placement of the movies in the genre "adventure" in red.

6.4.1 Rating Prediction The task of collaborative filtering consists of predicting the value of unknown ratings. To do that, we split the set of ratings into a training set and a test set. The test set contains 2% of all known ratings. Using the graph consisting of edges in the training set, we compute the signed Laplacian matrix, and use its inverse as a kernel for rating prediction. The collaborative filtering task consists of predicting the rating of edges.

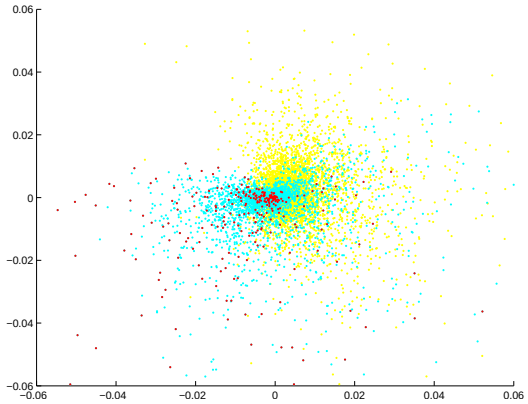
6.4.2 Algorithms In the setting of collaborative filtering, the graph $G = (V, E, A)$ is bipartite, containing user-vertices, item-vertices, and edges between them. In the MovieLens 100k corpus, ratings are given on an integer scale from 1 to 5. We scale these ratings to be evenly distributed around zero, by taking the mean of user and item ratings. Let $R_{ij} \in \{1, 2, 3, 4, 5\}$ be the rating by user i of item j, then we define

$$A_{ij} = A_{ji} \quad = \quad R_{ij} - (\mu_i + \mu_j)/2$$

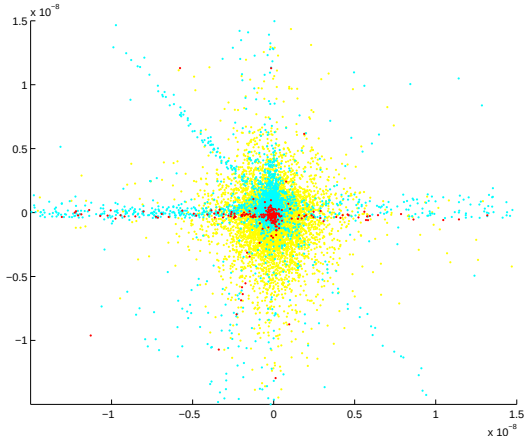
Where μ_i and μ_j are user and item means, and d_i and d_j are the number of rating given by each user or received by each item. The resulting adjacency matrix A is symmetric. We now describe the rating prediction algorithms we evaluate.

Mean As the first baseline algorithm, we use $(\mu_i + \mu_j)/2$ as the prediction.

Rank reduction As another baseline algorithm, we perform rank reduction on the symmetric matrix A by truncation of the eigenvalue decomposition $A = U\Lambda U^T$ to the k first eigenvectors and eigenvalues, giving $A_{(k)} = U_{(k)}\Lambda_{(k)}U_{(k)}^T$. We vary the value of k between 1 and 120. As a prediction for the user-item pair (i, j), we use the corresponding value in $A_{(k)}$. To that value, we add $(\mu_u + \mu_i)/2$ to scale the value back to the MovieLens rating scale.



(a) Adjacency matrix A



(b) Signed Laplacian matrix $\bar{L}$

Figure 8: The users and items of the MovieLens 100k corpus plotted using (a) eigenvectors of A with largest eigenvalue and (b) eigenvectors of $\bar{L}$ with smallest eigenvalue. Users are shown in yellow, movies in blue, and movies in the genre "adventure" in red. Edges are omitted for clarity. In the Laplacian plot, movies are clustered along lines through the origin, justifying the usage of the cosine similarity as described below.

Signed resistance distance kernel We use the signed Laplacian kernel $\bar{K} = \bar{L}^+$ for prediction. We compute the kernel on a rank-reduced eigenvalue decomposition of $\bar{L}$. As observed in Figure 8, clusters of vertices tend to form lines through the origin. Therefore, we suspect that the distance of a single point to the origin does not matter to rating prediction, and we use

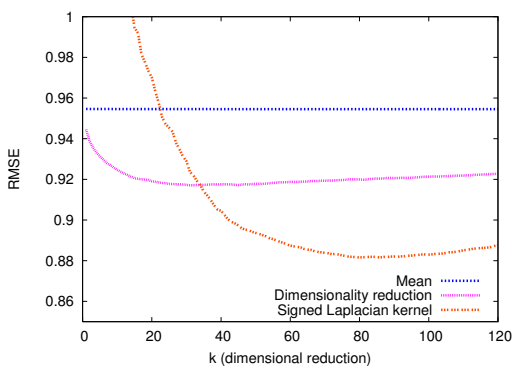


Figure 9: Root mean squared error of the rating prediction algorithms in function of the reduced rank k. Lower values denote better prediction accuracy.

the cosine similarity in the space spanned by the square root of the inverted Laplacian. This observation corresponds to established usage for the unsigned Laplacian graph kernel [7].

Given the signed Laplacian matrix $\bar{L}$ with eigendecomposition $\bar{L} = U\Lambda U^T$, we set $M = U\Lambda^{-1/2}$, noting that $K = MM^T$. Then, we normalize the rows of M to unit length to get the matrix N using $N_i = M_i/|M_i|$. The cosine distance between any vertices i and j is then given by $N_i N_j^T$, and NN^T is our new kernel. As with simple rank reduction, we scale the result to the MovieLens rating scale by adding $(\mu_u + \mu_i)/2$.

The evaluation results are shown in Figure 9. We observe that the signed Laplacian kernel has higher rating prediction accuracy than simple rank reduction and than the baseline algorithm. We also note that simple rank reduction achieves better accuracy for small values of k. All kernels display overfitting.

7 Electrical Networks

In this section, we complete the work of [18] by giving a derivation of the signed Laplacian matrix $\bar{L}$ in electrical networks with negative edges.

The resistance distance and heat diffusion kernels defined in the previous section are justified by physical interpretations. In the presence of signed edge weights, these interpretations still hold using the following formalism, which we describe in terms of electrical resistance networks.

A positive electrical resistance indicates that the potentials of two connected nodes will tend to each other: The smaller the resistance, the more both potentials approach each other. If the edge has negative weight, we can interpret the connection as consisting of a resistor of the corresponding positive weight in series with an *inverting amplifier* that guarantees its ends to have opposite voltage, as depicted in Figure 10. In other words, two nodes connected by a negative edge will tend to opposite voltages.

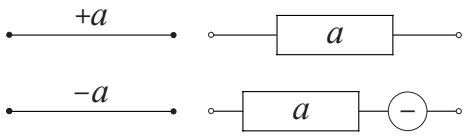


Figure 10: An edge with a negative weight is interpreted as a positive resistor in series with an inverting component.

Thus, a positive edge with weight $+a$ is modeled by a resistor of resistance a and a negative edge with weight $-a$ is modeled by a resistor of resistance a in series with a (hypothetical) electrical component that assures its ends have opposite electrical potential.

Electrical networks with such edges can be analysed in the following way. Let i be any node of an unsigned network. Then its electric potential is given by the mean of its neighbors' potentials, weighted by edge weights.

$$v_i = \left(\sum_{j\sim i} A_{ij} v_i\right)\left(\sum_{j\sim i} A_{ij}\right)^{-1}$$

If edges have negative weights, the inversion of potentials results in the following equation:

$$v_i = \left(\sum_{j\sim i} A_{ij} v_i\right)\left(\sum_{j\sim i} |A_{ij}|\right)^{-1}$$

Thus, electrical networks give the equation $\bar{D}v = Av$ and the matrices $\bar{D}^{-1}A$ and $\bar{D} - A$.

Because such an inverting amplifier needs the definition of a specific value of zero voltage, the resulting model loses one degree of freedom, explaining that in the general case $\bar{L}$ has rank one greater than L.

8 Discussion

The various example applications in this paper show that signed graphs appear in many diverse spectral graph mining applications, and that they can be approached by defining a variant of the Laplacian matrix $\bar{L}$. Although we used the notation $\bar{L} = \bar{D} - A$ throughout this paper, we propose the notation $L = \bar{D} - A$, since it does not interfere with unsigned graphs (in which $\bar{D} = D$) and because the matrix $D - A$ is less useful for data mining applications[2]. This signed extension not only exists for the combinatorial Laplacian L

[2] although $D - A$ is used for signed graphs in knot theory

but also for the two normalized Laplacians $I - D^{-1}A$ and $D^{-1/2}LD^{-1/2}$.

Spectral clustering of signed graphs is thus indeed possible using the intuitive measure of "cut" which counts positive edges between two clusters and negative edges inside each cluster. As for unsigned spectral clustering, the different Laplacian matrices correspond to ratio cuts and normalized cuts. For link prediction, we saw that the signed Laplacians can be used as kernels (since they are positive-semidefinite), and can replace graph kernels based on the adjacency matrix. This is especially true when the sign of edges is to be predicted. Finally, we derived the signed Laplacian using the application of graph drawing and electrical networks. These derivations should confirm that the definition $L = \bar{D} - A$ is to be preferred over $L = D - A$ for signed graphs.

In all cases, we observed that when the graph is unbalanced, zero is not an eigenvalue of the Laplacian and thus its eigenvector can be used directly unlike the unsigned case when the eigenvector of least eigenvalue is trivial and can be ignored. For graph drawing, this results in the loss of translational invariance of the drawing, i.e. the drawing is placed relative to a point zero. For electrical networks, this results in the loss of invariance under addition of a constant to electrical potentials, since the inverting amplification depends on the chosen zero voltage.

References

[1] M. Belkin and P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Advances in Neural Information Processing Systems*, pages 585–591, 2002.

[2] U. Brandes, D. Fleischer, and J. Lerner. Summarizing dynamic bipolar conflict structures. *Trans. on Visualization and Computer Graphics*, 12(6):1486–1499, 2006.

[3] P. Chebotarev and E. V. Shamis. On proximity measures for graph vertices. *Automation and Remote Control*, 59(10):1443–1459, 1998.

[4] F. Chung. *Spectral Graph Theory*. American Mathematical Society, 1997.

[5] M. Desai and V. Rao. A characterization of the smallest eigenvalue of a graph. *Graph Theory*, 18(2):181–194, 1994.

[6] I. S. Dhillon, Y. Guan, and B. Kulis. Kernel k-means: Spectral clustering and normalized cuts. In *Proc. Int. Conf. Knowledge Discovery and Data Mining*, pages 551–556, 2004.

[7] F. Fouss, A. Pirotte, J.-M. Renders, and M. Saerens. Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation. *Trans. on Knowledge and Data Engineering*, 19(3):355–369, 2007.

[8] C. Gkantsidis, M. Mihail, and E. Zegura. Spectral analysis of Internet topologies. In *Proc. Joint Conf. IEEE Computer and Communications Societies*, pages 364–374, 2003.

[9] K. Goldberg, T. Roeder, D. Gupta, and C. Perkins. Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval*, 4(2):133–151, 2001.

[10] P. Hage and F. Harary. *Structural Models in Anthropology*. Cambridge University Press, 1983.

[11] F. Harary. On the notion of balance of a signed graph. *Michigan Math.*, 2(2):143–146, 1953.

[12] Y. Hou. Bounds for the least Laplacian eigenvalue of a signed graph. *Acta Mathematica Sinica*, 21(4):955–960, 2005.

[13] T. Ito, M. Shimbo, T. Kudo, and Y. Matsumoto. Application of kernels to link analysis. In *Proc. Int. Conf. on Knowledge Discovery in Data Mining*, pages 586–592, 2005.

[14] D. J. Klein and M. Randić. Resistance distance. *Mathematical Chemistry*, 12(1):81–95, 1993.

[15] R. Kondor and J. Lafferty. Diffusion kernels on graphs and other discrete structures. In *Proc. Int. Conf. on Machine Learning*, pages 315–322, 2002.

[16] Y. Koren, L. Carmel, and D. Harel. ACE: A fast multiscale eigenvectors computation for drawing huge graphs. In *Symposium on Information Visualization*, pages 137–144, 2002.

[17] J. Kunegis, A. Lommatzsch, and C. Bauckhage. The Slashdot Zoo: Mining a social network with negative edges. In *Proc. Int. World Wide Web Conf.*, pages 741–750, 2009.

[18] J. Kunegis, S. Schmidt, C. Bauckhage, M. Mehlitz, and S. Albayrak. Modeling collaborative similarity with the signed resistance distance kernel. In *Proc. European Conf. on Artificial Intelligence*, pages 261–265, 2008.

[19] M. Lien and W. Watkins. Dual graphs and knot invariants. *Linear Algebra and its Applications*, 306(1):123–130, 2000.

[20] U. v. Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

[21] P. Massa and P. Avesani. Controversial users demand local trust metrics: an experimental study on epinions.com community. In *Proc. American Association for Artificial Intelligence Conf.*, pages 121–126, 2005.

[22] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems*, pages 849–856, 2001.

[23] K. E. Read. Cultures of the Central Highlands, New Guinea. *Southwestern J. of Anthropology*, (1):1–43, 1954.

[24] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.