

We now work towards Theorem 3.10, which characterizes the graphs whose deformed graph Laplacian has an eigenvalue of modulus < 1 . Let us observe that, by Proposition 3.7, it is no loss of generality to consider each connected component separately, i.e., assume that the graph is connected. Moreover, let d be the sum of all the degrees of the vertices in the graph, and let d/n be the average degree. Recall that a cycle is a closed path of length ≥ 3 . It can be proved that:

1. The graph contains no cycles (i.e. it is a tree) if and only if $d/n < 2$;
2. The graph contains precisely one cycle if and only if $d/n = 2$;
3. The graph contains more than one cycle if and only if $d/n > 2$.

With this in mind, Theorem 3.10 means that there are eigenvalues of $M(t)$ strictly within the unit circle if and only if some connected components of the underlying graph has two or more cycles. For networks coming from real-life applications, this is in a sense a generic condition, in that in almost all cases of practical interest to network scientists it happens to be true.

Theorem 3.10. *Let $A \in \mathbb{R}^{n \times n}$ be the adjacency matrix of a simple undirected graph, and let $M(t)$ be the associated deformed graph Laplacian. Then, there exists a finite eigenvalue λ , with $|\lambda| < 1$, of $M(t)$ if and only if the graph of A has at least one connected component whose average degree is > 2 .*

Proof. By Proposition 3.7, we can assume without loss of generality that the graph of A is connected.

Suppose first that the average degree is < 2 . Then, the graph of A is a tree, and the statement follows from Theorem 3.8 or, more directly, from Corollary 3.11. Similarly, if the average degree is precisely 2, then the statement follows by Lemma 3.8 and by computing the eigenvalues of the deformed graph Laplacian of a cycle graph (this task is left as a homework exercise).

To conclude the proof, suppose that the average degree of the graph of A is > 2 . Let $f(t)$ denote $\det M(t)$ for $0 \leq t \leq 1$. We will argue that $f(0) = 1$, $f(1) = 0$, and $f'(1) > 0$, implying by elementary analysis the existence of $\lambda \in (0, 1)$ such that $f(\lambda) = 0$.

That $f(0) = 1$ and $f(1) = 0$ is an immediate consequence of $M(0) = I$ and $M(1) = \Delta - A =: L$. We have $d = \sum_i \deg_i > 2n$. Moreover,

$$f'(t) = \frac{\partial \det M(t)}{\partial t} = \text{tr}(\text{adj}M(t) \cdot M'(t)),$$

where $\text{adj}X$ denotes the matrix adjugate of X . Evaluating at $t = 1$,

$$f'(1) = \text{tr}(\text{adj}L \cdot (2\Delta - A - 2I)) = -2\text{tr}(\text{adj}L) + \text{tr}(\text{adj}L \cdot A),$$

where for the last equality we exploited the relation $\text{adj}L \cdot L = 0$. Let $\lambda_1 \geq \dots \geq \lambda_{n-1} \geq \lambda_n = 0$ be the eigenvalues of L and set $p = \prod_{i=1}^{n-1} \lambda_i$. It is known [?] that $p > 0$ if the graph of A is connected. It is straightforward to show that

$$\text{adj}L = \frac{p}{n} \mathbf{1}\mathbf{1}^T$$

implying $\text{tr}(\text{adj}L) = p$ and $\text{tr}(\text{adj}L \cdot A) = pd/n$. Hence, $f'(1) = p(\frac{d}{n} - 2) > 0$. \square

3.4.5 The Radius of Convergence of the Nonbacktracking Generating Function

The power series $\sum_{r=0}^{\infty} p_r(A)t^r$ makes sense mathematically for any $t \in \mathbb{C}$; although, as mentioned in section 3.4.1, for network analysis it is natural to focus on $t \in (0, 1) \subset \mathbb{R}$. In this subsection we study the radius of convergence of this power series to its generating function $(1 - t^2)M(t)^{-1}$. We note that it may happen that 1 or -1 are within the radius of convergence but are also eigenvalues of $M(t)$, so that the latter is not invertible at $t = 1$ or $t = -1$. In this case, and with slight abuse of notation, when talking of $(1 - t^2)M(t)^{-1}$ for $t = 1$ or $t = -1$ we tacitly mean the appropriate limit of this expression for $t \rightarrow 1$ or $t \rightarrow -1$. (The existence of the limit in this scenario follows from the analysis below and standard results on rational functions).

First, we note that, by construction, elementwise it holds that

$$p_k(A) = |p_k(A)| \leq |A^k| = A^k.$$

Hence, $|t| < \rho(A)^{-1}$, where $\rho(A)$ is the spectral radius of A , surely suffices for convergence, since $\sum_{k=0}^{\infty} A^k t^k$ has radius of convergence $\rho(A)^{-1}$.

However, this condition is sufficient but not necessary, as shown by the star graph example of section 3.4.1, where $\rho(A) = \sqrt{n-1}$ yet we have convergence for all t . (Indeed, although this is beyond the scope of these notes, it can be proved that if the graph is not the empty graph then the radius of convergence of the power series $\sum_{r=0}^{\infty} p_r(A)t^r$ is always *strictly* larger than $\rho(A)^{-1}$, i.e., the range of the NBTW parameter t is always a proper superset of the range of the Katz parameter α .) More generally, if the graph A is a

tree (or a forest) so that there are no cycles, then $p_k(A) = 0$ for large enough values of k . It follows that $\Phi(A, t)$ is a polynomial in t , and the series converges for all t . This implies that $M(t)$ cannot have any finite eigenvalues other than ± 1 ; and both 1 and -1 must be eigenvalues by Proposition 3.6 (and noting that any tree is bipartite). his observation yields the following corollary on the spectral properties of $M(t)$ in the case of a forest.

Corollary 3.11. *Suppose that A is the adjacency matrix of a forest, and let $M(t)$ be the associated deformed graph Laplacian. Then, $M(t)$ has the only finite eigenvalues 1 and -1 . Moreover, 1 and -1 are both semisimple eigenvalues.*

For a general A , our analysis is based on the properties of the deformed graph Laplacian $M(t)$ as a matrix polynomial. The following technical lemma will be useful. It follows from the conditional converse to Abel's Theorem on power series.

Lemma 3.12. *For any $z \in \mathbb{C}$ let $\sum_{k=0}^{\infty} a_k z^k$ be a power series with nonnegative real coefficients, i.e., $a_k \geq 0 \forall k \in \mathbb{N}$. Suppose that the power series converges to the rational function $p(z)/q(z)$, with $p(z), q(z) \in \mathbb{R}[z]$ coprime polynomials, with radius of convergence $r > 0$. Then, $q(r) = 0$.*

We are now ready to conclude our analysis and state (without a full proof) the result on the convergence of the matrix power series $\sum_r p_r(A)t^r$. It turns out that it is determined by a particular eigenvalue of $M(t)$.

Theorem 3.13. *Let A be the adjacency matrix of a simple, undirected, graph. Let $M(t) = I - tA + t^2(\Delta - I)$ be the associated deformed graph Laplacian, and let $\ell_n(t)$ be the n th invariant polynomial of $M(t)$, as defined in Theorem ??.*

The radius of convergence of the power series in Theorem 3.5 is equal to $|\chi|$ where χ is the smallest (in modulus) zero of

$$r(t) := \frac{\ell_n(t)}{1 - t^2}, \quad (17)$$

while $\chi := \infty$ if such a function does not vanish.

Moreover, let λ be the smallest (in modulus) eigenvalue of the matrix polynomial $M(t)$.

1. *There exists $\mu \in (0, 1] \subset \mathbb{R}$ such that (i) $\mu = |\lambda|$ and (ii) μ is an eigenvalue of $M(t)$.*

2. If $\mu < 1$, then $|\chi| = \mu$.
3. If $\mu = 1$ is a semisimple eigenvalue of $M(t)$, then the underlying graph is a forest and $\chi = \infty$.
4. If $\mu = 1$ is a defective eigenvalue of $M(t)$, then $|\chi| = \mu = 1$.

We note that, as a consequence of Theorem 3.13 and the fact that $|t| < \rho(A)^{-1}$ is a sufficient condition for convergence of the power series $\sum_k p_k(A)t^k$, the following corollary gives a lower bound for μ which is tighter than zero.

Corollary 3.14. *Let $M(t)$ be the deformed graph Laplacian associated with a graph with adjacency matrix A whose spectral radius is $\rho(A)$. Then, every eigenvalue λ of $M(t)$ satisfies $|\lambda| \geq \rho(A)^{-1}$.*

Theorem 3.13 reduces to the following simpler form assuming that $\mu < 1$, which by Theorem 3.10 is equivalent to assuming that the underlying graph has at least one connected component with average degree > 2 (or equivalently, at least one connected components that contains two or more cycles). We note that this result therefore covers a general case that is likely to be encountered frequently in practice.

Proposition 3.15. *Suppose that the matrix polynomial $M(t) = I - tA + t^2(\Delta - I)$ has an eigenvalue λ with $|\lambda| < 1$. Then there exists a positive real number μ such that μ is the smallest (in modulus) eigenvalue of $M(t)$. Moreover, the radius of convergence of the power series $\sum_{k=0}^{\infty} t^k p_k(A)$ is equal to μ .*

Furthermore, if the graph of A is connected, μ is a simple eigenvalue of $M(t)$ and every other finite eigenvalue λ satisfies $\mu < |\lambda| \leq 1$.

Proof. Except for the last sentence, the statement is an immediate corollary of Theorem 3.13. The proof of the last sentence is omitted from these notes (but can be studied by students interested, as a project, in studying the deformed graph Laplacian further). □

3.5 Exercises

1. Prove that, in a simple graph, the degree of a vertex is equal to the number of closed paths of length 2 starting from and ending on that vertex.

2. Prove that the “signless graph Laplacian”, defined as $\Delta + A$, is a positive semidefinite matrix.
3. Improve Corollary 3.14 by showing that, in fact, every eigenvalue λ of $M(t)$ satisfies $\lambda > \rho(A)^{-1}$. Hint: Use an argument similar to the proof of Theorem 3.9.
4. Prove Proposition 3.7.

4 Clustering

The problem of clustering is, given a graph $G = (V, E)$ that of partitioning $V(G)$ into k subsets (in the most common and basic case, $k = 2$) in a “good” way. What “good” means is, as in many other instances in this course, application dependent, and indeed one may think of situations where the requirements are opposite to each other.

4.1 The minimum k -cut problem

One possibility is to aim, in some sense, at the opposite of the natural partition of a bipartite graph. In a bipartite graph, the partition is such that all the edges connect one community with the other, and no edge is internal to one of the communities. If we want, instead, to identify a community within a network, it makes more sense to require that as many edges as possible are internal within each community, and as few as possible connect one community with a different one. If the number of communities to be identified is k , this is what in graph theory is known as the *minimum k -cut problem* (a cut is a partition of the vertices). For a simple graph, this is to minimize over all partitions of V into k disjoint non-empty subsets V_i , the so called *cut function*

$$f(V_1, \dots, V_k) = \sum_{1 \leq i \neq j \leq k} \sum_{v \in V_i, w \in V_j} 1.$$

(This kind of problem is easily extendable to the case of weighted graph: rather than minimizing the count of edges that connect distinct partitions, in that scenario we could simply minimize the summation of the weights of such edges. However, for simplicity, in this course we will focus on the unweighted case.)

The minimum k -cut problem is an instance of a discrete optimization problem. It is known that its complexity is $O(n^{k^2})$, where $n = \#V$ is the number of vertices in the graph. Even for $k = 2$, this means $O(n^4)$ which for many real-life applications is impractical. For this reason, we will look for a relaxation, and we will see that once again we will be led towards a technique based on spectral tools, i.e., eigenvalues and eigenvectors. We will do this for $k = 2$.

Example 4.1. *Let us consider for instance the minimum 2-cut problem for $G = (V, E)$ where $V = [4]$ and $E = \{12, 21, 23, 32, 24, 42, 34, 43\}$. Since*

exchanging the roles of V_1 and V_2 leaves the cut function unchanged, let us assume with no loss of generality that $\#V_1 \leq \#V_2$, which is equivalent to $1 \leq \#V_1 \leq 2$. There are therefore precisely $\binom{4}{1} + \binom{4}{2} = 10$ such choices for V_1 . Note that, when $\#V_1 = 1$, the value of the cut function is precisely equal to the degree of the element of V_1 . For the cases where $\#V_1 = 2$, we can manually count the value of the cut function; note that, again by the symmetry after switching roles of V_1 and V_2 , we only need to check half of the cases with $\#V_1 = \#V_2 = 2$. For example, we may assume that $1 \in V_1$. We summarize the resulting values below.

V_1	$f(V_1, V_2)$
$\{1\}$	2
$\{2\}$	6
$\{3\}$	4
$\{4\}$	4
$\{1, 2\}$	4
$\{1, 3\}$	6
$\{1, 4\}$	6

We conclude that, up to switching V_1 and V_2 , there is a unique solution to the minimum 2-cut problem on this example, and it is $V_1 = \{1\}, V_2 = \{2, 3, 4\}$.

Note that, if the graph has k connected components (or more), the minimum k -cut problems has at least one trivial optimum where the function value is 0; indeed, this is also true only if the graph has k or more connected components. For this reason, this problem is more interesting when the underlying graph is connected, and we will make this assumption below.

As a first step towards the sought method is to re-express the minimum 2-cut problem as an optimization problem over a certain set of vectors. Indeed, we can represent a partition $V = V_1 \cup V_2$, $V_1 \cap V_2 = \emptyset$ with a vector $v \in \{-1, 1\}^n$ by defining the vector v componentwise as follows:

$$v_i = \begin{cases} 1 & \text{if } i \in V_1; \\ -1 & \text{if } i \in V_2. \end{cases}$$

With the exception of the vector of all ones e and of $-e$ (which do not correspond to a partition as V_1 and V_2 must be non-empty) each such vector uniquely represents a partition. There are $2^n - 2$ such vectors: a total of 2^n vectors with entries in $\{-1, 1\}$, minus the two invalid ones. Moreover, observe that changing the sign of such a vector corresponds to switching V_1 with V_2 , which effectively represents the same partition because it certainly cannot be distinguished by the original partition in terms of the number of edges that connect the two communities. Hence, in practice that are $2^{n-1} - 1$ possible minimisers to check. (Compare with the Example above: for that graph, $n = 4$ and hence $2^{n-1} - 1 = 7$.)

Lemma 4.2. *The cut function to be minimized in the minimum 2-cut problem can be expressed as a function of the vector v that represents, as above, a partition of V into 2 disjoint subsets in the following way:*

$$f(v) = \frac{1}{2}v^T Lv,$$

where L is the graph Laplacian of the underlying simple graph.

Proof. Let us first observe that for $k = 2$ the cut function simplifies to $f(V_1, V_2) = 2 \sum_{i \in V_1, j \in V_2, ij, ji \in E} 1$. Observe that $v_i - v_j$ is equal to 2 if $i \in V_1, j \in V_2$, to -2 if $i \in V_2, j \in V_1$, and to 0 if i, j belong to the same subset V_i ($i = 1, 2$).

Now, recall that $v^T Lv = \sum_{ij, ji \in E} (v_i - v_j)^2$ (see the observations on page 13). By the above comments, this is in turn equal to

$$v^T Lv = 4 \sum_{ij, ji \in E, v_i, v_j \text{ not in the same subset of } V} 1 = 2f(V_1, V_2).$$

□

Therefore, the minimum 2-cut problem is equivalent to the problem of minimizing $v^T Lv$ over all vectors v with all entries equal to ± 1 , excluding $\pm e$.

4.2 The Fiedler vector

In the previous section, we have reformulated the minimum 2-cut problem into an integer optimization problem. What makes the problem difficult, intuitively, is the constraint that all entries of the vector v must be either -1 or $+1$. A common trick in optimization, when a constraint makes it difficult to solve the problem, is to relax it. We could do this by asking that v is *any* real vector, maybe normalized so that its norm is fixed, say, $v^T v = 1$. However, we already know what happens by the Rayleigh theorem: indeed, this theorem implies that, since the least eigenvalue of L is $\lambda_n = 0$,

$$\min_{v^T v=1} v^T L v = 0;$$

moreover, the argument minimum is the corresponding eigenvector, which we know being (forgetting for a moment the normalization) to be proportional to e . Since all the entries of e are equal to 1, this is not very useful for clustering. We will go round this issue by looking at the next eigenpair, that indeed has a name that honours the mathematician M. Fiedler who first observed that this could be done.

Definition 4.1. *Let G be a simple, connected graph, and denote by L the corresponding graph Laplacian. The second smallest eigenvalue of L is called the spectral gap of the graph, and a corresponding eigenvector is called a Fiedler vector of the graph.*

The connection with our clustering problem can be explained as follows: the original problem was too difficult to be solved efficiently, but the relaxed problem has a completely uninformative solution. As we had totally relaxed the original constraint, and since the issue is that the new solution is a vector with all equal component, let us introduce a new (but milder) constraint, by optimizing over all real vectors v that are orthogonal to e . This guarantees that v must have some positive and some negative entries (because their sum must be 0). Another theorem in classical matrix analysis, the Courant-Fischer theorem, tells us that

$$\min_{v^T v=1, e^T v=0} v^T L v = \lambda_{n-1},$$

where λ_{n-1} is the spectral gap, and that moreover the argument minimum is the Fiedler vector f (normalized to have norm 1). Of course, f will not

have all components equal to ± 1 , but assuming that they are all nonzero, we can for example assign node i to V_1 if $f_i > 0$ and to V_2 if $f_i < 0$. Note that, although the Fiedler vector (being an eigenvector) is only defined up to a nonzero constant, the method is coherent: indeed, multiplying the Fiedler vector by a positive constant does not change the signs, and hence it leaves the assignments unchanged; whereas multiplying the Fiedler vector by a negative constant changes *all* the signs, and hence it reverts all the assignments, i.e., it switches the roles of V_1 and V_2 . If there are zero components in f , one could assign them at random, or (if they are not too many) one could try all the possibilities, and choose the one that minimizes the cut function (having already assigned all the nodes corresponding to nonzero component, one can hope that this brute force approach is not very expensive).

Example 4.3. *Let us go back to the graph $G = (V, E)$ where $V = [4]$ and $E = \{12, 21, 23, 32, 24, 42, 34, 43\}$. The graph Laplacian of G is*

$$L = \begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 3 & -1 & -1 \\ 0 & -1 & 2 & -1 \\ 0 & -1 & -1 & 2 \end{bmatrix}.$$

We can compute its eigenvalues, which turn out to be 1, 2, 3, 4. Therefore, the spectral gap is 1. Solving $Lf = f$ yields, for any nonzero constant α , $f = \alpha [2 \ 0 \ -1 \ -1]^T$. Thus, we assign 1 to V_1 and 3, 4 to V_2 . With the proposed algorithm, 2 can be either assigned to V_1 or V_2 : note that the actual optimum corresponds to $2 \in V_2$ and the rest as prescribed by the Fiedler eigenvector.

Example 4.4. *Let now G obtained as follows: take the cycle graph with 7 nodes (labelled 1 through 7) and the cycle graph with 3 nodes (labelled 8 through 10) and connect them with a single edge joining, say, vertices 1 and*

8. Then,

$$L = \begin{bmatrix} 3 & -1 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & -1 & 2 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 2 \end{bmatrix};$$

it is in this case convenient to rely on approximated computation. Invoking for example MATLAB, we can compute that the spectral gap is approximately $\lambda_{n-1} \simeq 0.2375$. Here is the approximate computation of the Fiedler vector:

$$f \simeq \begin{bmatrix} -0.0521 \\ 0.1147 \\ 0.2543 \\ 0.3335 \\ 0.3335 \\ 0.2543 \\ 0.1147 \\ -0.3734 \\ -0.4897 \\ -0.4897 \end{bmatrix};$$

thus, the Fiedler spectral clustering gets very close to (but is not quite able to) distinguish the two cycles within this graph. Note that f_1 is negative, but quite small.

The example above is illustrative of one feature of Fiedler clustering: we have added the constraint $v^T e = 0$: if v still had all entries in $\{-1, 1\}$, and G had an even number of vertices, this would imply that V_1 and V_2 have the same number of vertices. This can intuitively explain why Fiedler clustering failed to identify the two cycles (which is the minimum cut: indeed the cut function is 1 for that partition, and since the graph is connected, it can never assume the value 0): due to the new constraint, it somehow seeks a

balance between minimum cut and equal distribution of vertices. This is maybe unwanted, but it is the price to pay to make the problem much more tractable computationally.

We conclude this subsection by giving a result from the original paper by M. Fiedler; the proof that we give is not Fiedler's original one, but a simplification of an argument first proposed by J. Demmel.

Theorem 4.5 (Fiedler 1975). *Let $G = (V, E)$ be a simple connected graph and assume the Fiedler vector f does not have any zero entry. Denote by G_1, G_2 the two subgraphs of G induced by V_1, V_2 where $V = V_1 \cup V_2$ is the partition obtained by the method above described. Then G_1 and G_2 are connected.*

Proof. Suppose for a contradiction that G_1 is not connected. Then the graph Laplacian L of G , up to a permutation of the nodes so that the indices of the first connected component of G_1 come first and those of G_2 come last, must have the form

$$L = \begin{bmatrix} L_{11} & 0 & -A_{13} \\ 0 & L_{22} & -A_{23} \\ -A_{13}^T & -A_{23}^T & L_{33} \end{bmatrix};$$

let us also partition the Fiedler vector accordingly as

$$f = \begin{bmatrix} x \\ y \\ -z \end{bmatrix}$$

with $x, y, z > 0$ componentwise. Moreover, denoting by $\phi > 0$ the spectral gap of G , we have the equations

$$\begin{cases} L_{11}x + A_{13}z = \phi x, \\ L_{22}y + A_{23}z = \phi y, \\ -A_{13}^T x - A_{23}^T y - L_{33}z = -\phi z. \end{cases}$$

We recall a (consequence of) the *Cauchy interlacing theorem*: if M is a symmetric matrix with eigenvalues $\lambda_1 \geq \dots \geq \lambda_n$ and let S be any $m \times m$ principal (i.e. obtained by selecting the same subset of rows and columns) submatrix of M , with eigenvalues $\mu_1 \geq \dots \mu_m$. Then, for all $j = 1, \dots, m$, it holds that

$$\lambda_{n-m+j} \leq \mu_j \leq \lambda_j.$$

Note, in particular, that (specializing to $j = m - 1$) it must be $\lambda_{n-1} \leq \mu_{m-1}$. Specializing to graph Laplacians, this can be interpreted that the any principal submatrix of a graph Laplacian cannot have more than one eigenvalue less than the spectral gap of the graph Laplacian under consideration.

Now, observe that, by the Rayleigh theorem, denoting by $\mu_{\min}(L_{11})$ the smallest eigenvalue of L_{11} ,

$$\mu_{\min}(L_{11}) = \min_{v \neq 0} \frac{v^T L_{11} v}{v^T v} \leq \frac{x^T L_{11} x}{x^T x}$$

which implies that

$$x^T x \mu_{\min}(L_{11}) \leq x^T L_{11} x = \phi x^T x - x^T A_{13} z < \phi x^T x.$$

To explain the last step, observe that $A_{13} \geq 0$, and $\neq 0$ (else, G is disconnected!). Since $z > 0$, it follows that $-A_{13} z \leq 0$ and $\neq 0$. On the other hand, $x > 0$, and hence, $-x^T A_{13} z < 0$. We conclude that $\mu_{\min}(L_{11}) < \phi$.

Analogously, we can show that

$$y^T y \mu_{\min}(L_{22}) \leq y^T L_{22} y = \phi y^T y - y^T A_{23} z < \phi y^T y \Rightarrow \mu_{\min}(L_{22}) < \phi.$$

Hence, the matrix

$$\begin{bmatrix} L_{11} & 0 \\ 0 & L_{22} \end{bmatrix}$$

is a principal submatrix of L , of size at most $n - 1$, having at least two eigenvalues *strictly* smaller than ϕ : this contradicts the Cauchy interlacing theorem that implies that the 2 smallest eigenvalues of L (among which is ϕ) are bounded above by the second smallest eigenvalue of any submatrix. \square

4.3 Clustering via adjacency matrix eigenvectors

In this subsection, we describe a modification of the Fiedler algorithm which is based on a similar idea, but employs the adjacency matrix A . This time, we impose orthogonality with the Perron-Frobenius eigenvector A , and look at the eigenvector associated with the second largest eigenvalue, say, s . Then, once again, we can look at the sign pattern in s to cluster the nodes.

Example 4.6. Let us consider the same graph G as in Example 4.4. Its adjacency matrix is

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{bmatrix};$$

The second largest eigenvalue of A is $\lambda_2 \simeq 1.8692$. Here is the approximate computation of the corresponding eigenvector:

$$s \simeq \begin{bmatrix} 0.1206 \\ 0.2524 \\ 0.3512 \\ 0.4041 \\ 0.4041 \\ 0.3512 \\ 0.2524 \\ -0.2794 \\ -0.3214 \\ -0.3214 \end{bmatrix};$$

interestingly, on this example this method does identify the minimum 2-cut.

Having to be orthogonal to a positive (or nonnegative) eigenvector, the second largest eigenvector must always have positive and negative entries. An advantage of this method is that it can be generalized to directed graphs. In that scenario, rather than an eigendecomposition, it is convenient to consider a singular value decomposition $A = U\Sigma V^T = \sum_{i=1}^n \sigma_i u_i v_i^T$. The leading singular vectors u_1, v_1 are the Perron-Frobenius eigenvectors of AA^T and $A^T A$, respectively: as a consequence, they are nonnegative. The next singular vectors u_2, v_2 must be orthogonal to u_1, v_1 respectively, and thus have entries of both positive and negative signs. The resulting SVD clustering are generally

distinct: seeing A as a matrix whose columns represent nodes as *authorities* (sources that are followed by many other nodes) whose rows represent nodes as *hubs* (sources that follow many other nodes), then the sign pattern in u_2 clusters hubs while the sign pattern in v_2 clusters authorities. (If $A = A^T$, the hubs and authorities SVD clustering coincide, and are the same as the eigenvector clustering described above.)