Eigenvector Centrality

Jeff Chak Fu WONG

Department of Mathematics
The Chinese University of Hong Kong

2526 EDB HSMMC Math Modelling Advanced Workshop

Eigenvector Centrality

Eigenvector centrality measures the importance of a node in a network.

It assigns relative scores to all nodes in the network based on the principle that connections to high-scoring nodes contribute more to the score of the node than equal connections to low-scoring nodes.

The eigenvector centrality defined in this way computes each node a centrality that depends both on the number and the quality of its connections:

having a large number of connections still counts for something, but a node (or an actor) with a smaller number of high-quality contacts may outrank one with a larger number of mediocre contacts.

The eigenvector centrality of a node is proportional to the centrality scores of its neighbors.

Assume x_i gives the importance of node i, and $N(i)$ gives set of neighbours of i:

$$x_i = \frac{1}{\lambda} \sum_{j \in N(i)} x_j$$

where

$$N(i) = \{\, j| \;\, a_{ij} = 1\}.$$

Important if you have many connections (of some importance), or a few but very important connections!

How can we write this in matrix notation?

Note that we have

$$\sum_{j \in N(i)} x_j = \sum_{j=1}^{n} a_{ij} x_j$$

where $\mathbf{x}$ is a vector of all centrality scores.

Recall that

	n_1	n_2	$\cdots$	n_j	$\cdots$	n_n
n_1	0	a_{12}		a_{1j}		a_{1n}
n_2	a_{21}	0				a_{2n}
$\vdots$			$\ddots$			
n_i	a_{i1}	a_{i2}		a_{ij}		a_{in}
$\vdots$						
n_n	a_{n1}	a_{n2}		a_{nj}		0

Let n_i and n_j be the ith and jth nodes.
For an undirected graph, we have

$$\begin{cases} a_{ij} = 1 & \text{when there is a connection between } n_i \text{ and } n_j; \\ a_{ij} = 0 & \text{when there is no connection between } n_i \text{ and } n_j; \\ a_{ii} = a_{jj} = 0 & \text{when the connection does not exits;} \end{cases}$$

and the adjacent matrix is symmetric, i.e.,

$$a_{ij} = a_{ji}.$$

We can express this relation with the following formula:

$$x_i = \frac{1}{\lambda} \sum_{j=1}^{n} a_{ij} x_j,$$

where

- a_{ij} is the adjacency matrix of the network graph, i.e., $A = [a_{ij}]$,
- n is the number of nodes in this network, and
- λ is a constant that denotes each node's share from the sum of its neighbors' centrality scores.

Using graph notations with the ith node, the centrality score, x_i is proportional to the sum of the scores of all nodes which are connected to it, i.e.,

$$\begin{aligned} x_i &= \frac{1}{\lambda} \sum_{j=1}^{n} a_{ij} x_j \\ &= \frac{1}{\lambda} \underbrace{(a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{ij}x_j + \cdots + a_{in}x_n)}_{\text{number of } n \text{ terms}} \end{aligned}$$

	n_1	n_2	$\cdots$	n_j	$\cdots$	n_n
n_1	0	a_{12}		a_{1j}		a_{1n}
n_2	a_{21}	0				a_{2n}
$\vdots$			$\ddots$			
n_i	a_{i1}	a_{i2}		a_{ij}		a_{in}
$\vdots$						
n_n	a_{n1}	a_{n2}		a_{nj}		0

Let n_i and n_j be the ith and jth nodes.

For an undirected graph, we have

$$\begin{cases} a_{ij} = 1 & \text{when there is a connection between } n_i \text{ and } n_j; \\ a_{ij} = 0 & \text{when there is no connection between } n_i \text{ and } n_j; \\ a_{ii} = a_{jj} = 0 & \text{when the connection does not exits;} \end{cases}$$

and the adjacent matrix is symmetric, i.e.,

$$a_{ij} = a_{ji}.$$

What is x_i?

Let us use the following undirected graph to demonstrate an individual node's eigenvector centrality, x_i.

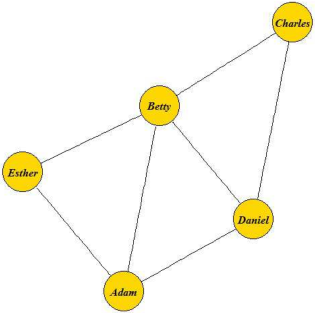


Figure 1: A Friendship Social Network

We need an adjacent matrix!

Friend		A	B	C	D	E
Adam	A	0	1	0	1	1
Betty	B	1	0	1	1	1
Charles	C	0	1	0	1	0
Daniel	D	1	1	1	0	0
Esther	E	1	1	0	0	0

	A	B	C	D	E
A	0	1	0	1	1
B	1	0	1	1	1
C	0	1	0	1	0
D	1	1	1	0	0
E	1	1	0	0	0

	A	B	C	D	E
A	0	1	0	1	1
B	1	0	1	1	1
C	0	1	0	1	0
D	1	1	1	0	0
E	1	1	0	0	0

Let us construct $x_i, i = 1, 2, 3, 4, 5$

$$
\begin{aligned}
x_i &= \frac{1}{\lambda} \sum_{j=1}^{5} a_{ij} x_j \\
&= \frac{1}{\lambda} \underbrace{(a_{i1}x_1 + a_{i2}x_2 + a_{i2}x_2 + a_{i3}x_3 + a_{i5}x_5)}_{\text{number of } five \text{ terms}}
\end{aligned}
$$

Here is a useful mapping:

i	Relabel
1	A
2	B
3	C
4	D
5	E

$$x_A = \frac{1}{\lambda}\left(a_{AB}x_B + a_{AC}x_C + a_{AD}x_D + a_{AE}x_E\right)$$
$$x_B = \frac{1}{\lambda}\left(a_{BA}x_A + a_{BC}x_C + a_{BD}x_D + a_{BE}x_E\right)$$
$$x_C = \frac{1}{\lambda}\left(a_{CA}x_A + a_{CB}x_B + a_{CD}x_D + a_{CE}x_E\right)$$
$$x_D = \frac{1}{\lambda}\left(a_{DA}x_A + a_{DB}x_B + a_{DC}x_C + a_{DE}x_E\right)$$
$$x_E = \frac{1}{\lambda}\left(a_{EA}x_A + a_{EB}x_B + a_{EC}x_C + a_{ED}x_D\right)$$

Friend		A	B	C	D	E
Adam	A	0	1	0	1	1
Betty	B	1	0	1	1	1
Charles	C	0	1	0	1	0
Daniel	D	1	1	1	0	0
Esther	E	1	1	0	0	0

Use the matrix multiplication:

$$\begin{bmatrix} x_A \\ x_B \\ x_C \\ x_D \\ x_E \end{bmatrix} = \frac{1}{\lambda} \begin{bmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_A \\ x_B \\ x_C \\ x_D \\ x_E \end{bmatrix}$$

$$\begin{bmatrix} x_A \\ x_B \\ x_C \\ x_D \\ x_E \end{bmatrix} = \frac{1}{\lambda} \begin{bmatrix} 0 & a_{AB} & a_{AC} & a_{AD} & a_{AE} \\ a_{BA} & 0 & a_{BC} & a_{BD} & a_{BE} \\ a_{CA} & a_{CB} & 0 & a_{CD} & a_{CE} \\ a_{DA} & a_{DB} & a_{DC} & 0 & a_{DE} \\ a_{EA} & a_{EB} & a_{EC} & a_{ED} & 0 \end{bmatrix} \begin{bmatrix} x_A \\ x_B \\ x_C \\ x_D \\ x_E \end{bmatrix}$$

$$
\begin{bmatrix} x_A \\ x_B \\ x_C \\ x_D \\ x_E \end{bmatrix} = \frac{1}{\lambda} \begin{bmatrix} 0 & a_{AB} & a_{AC} & a_{AD} & a_{AE} \\ a_{BA} & 0 & a_{BC} & a_{BD} & a_{BE} \\ a_{CA} & a_{CB} & 0 & a_{CD} & a_{CE} \\ a_{DA} & a_{DB} & a_{DC} & 0 & a_{DE} \\ a_{EA} & a_{EB} & a_{EC} & a_{ED} & 0 \end{bmatrix} \begin{bmatrix} x_A \\ x_B \\ x_C \\ x_D \\ x_E \end{bmatrix}
$$

$$
\begin{aligned}
x_A &= \frac{1}{\lambda} \left(a_{AB} x_B + a_{AC} x_C + a_{AD} x_D + a_{AE} x_E \right) \\
&= \frac{1}{\lambda} \left(\underbrace{a_{AB}}_{=1} x_B + \cancelto{=0}{a_{AC}} x_C + \underbrace{a_{AD}}_{=1} x_D + \underbrace{a_{AE}}_{=1} x_E \right) \\
&= \frac{1}{\lambda} \left(x_B + x_D + x_E \right)
\end{aligned}
$$

$$\begin{bmatrix} x_A \\ x_B \\ x_C \\ x_D \\ x_E \end{bmatrix} = \frac{1}{\lambda} \begin{bmatrix} 0 & a_{AB} & a_{AC} & a_{AD} & a_{AE} \\ a_{BA} & 0 & a_{BC} & a_{BD} & a_{BE} \\ a_{CA} & a_{CB} & 0 & a_{CD} & a_{CE} \\ a_{DA} & a_{DB} & a_{DC} & 0 & a_{DE} \\ a_{EA} & a_{EB} & a_{EC} & a_{ED} & 0 \end{bmatrix} \begin{bmatrix} x_A \\ x_B \\ x_C \\ x_D \\ x_E \end{bmatrix}$$

$$\begin{aligned} x_B &= \frac{1}{\lambda}\left(a_{BA}x_A + a_{BC}x_C + a_{BD}x_D + a_{BE}x_E\right) \\ &= \frac{1}{\lambda}\left(x_A + x_C + x_D + x_E\right) \end{aligned}$$

$$
\begin{bmatrix} x_A \\ x_B \\ x_C \\ x_D \\ x_E \end{bmatrix} = \frac{1}{\lambda} \begin{bmatrix} 0 & a_{AB} & a_{AC} & a_{AD} & a_{AE} \\ a_{BA} & 0 & a_{BC} & a_{BD} & a_{BE} \\ a_{CA} & a_{CB} & 0 & a_{CD} & a_{CE} \\ a_{DA} & a_{DB} & a_{DC} & 0 & a_{DE} \\ a_{EA} & a_{EB} & a_{EC} & a_{ED} & 0 \end{bmatrix} \begin{bmatrix} x_A \\ x_B \\ x_C \\ x_D \\ x_E \end{bmatrix}
$$

$$
\begin{aligned} x_C &= \frac{1}{\lambda} \left(a_{CA} x_A + a_{CB} x_B + a_{CD} x_D + a_{CE} x_E \right) \\ &= \frac{1}{\lambda} \left(x_B + x_D \right) \end{aligned}
$$

$$\begin{bmatrix} x_A \\ x_B \\ x_C \\ x_D \\ x_E \end{bmatrix} = \frac{1}{\lambda} \begin{bmatrix} 0 & a_{AB} & a_{AC} & a_{AD} & a_{AE} \\ a_{BA} & 0 & a_{BC} & a_{BD} & a_{BE} \\ a_{CA} & a_{CB} & 0 & a_{CD} & a_{CE} \\ a_{DA} & a_{DB} & a_{DC} & 0 & a_{DE} \\ a_{EA} & a_{EB} & a_{EC} & a_{ED} & 0 \end{bmatrix} \begin{bmatrix} x_A \\ x_B \\ x_C \\ x_D \\ x_E \end{bmatrix}$$

$$\begin{aligned} x_D &= \frac{1}{\lambda}\left(a_{DA}x_A + a_{DB}x_B + a_{DC}x_C + a_{DE}x_E\right) \\ &= \frac{1}{\lambda}\left(x_A + x_B + x_C\right) \end{aligned}$$

$$\begin{bmatrix} x_A \\ x_B \\ x_C \\ x_D \\ x_E \end{bmatrix} = \frac{1}{\lambda} \begin{bmatrix} 0 & a_{AB} & a_{AC} & a_{AD} & a_{AE} \\ a_{BA} & 0 & a_{BC} & a_{BD} & a_{BE} \\ a_{CA} & a_{CB} & 0 & a_{CD} & a_{CE} \\ a_{DA} & a_{DB} & a_{DC} & 0 & a_{DE} \\ a_{EA} & a_{EB} & a_{EC} & a_{ED} & 0 \end{bmatrix} \begin{bmatrix} x_A \\ x_B \\ x_C \\ x_D \\ x_E \end{bmatrix}$$

$$\begin{aligned} x_E &= \frac{1}{\lambda}\left(a_{EA}x_A + a_{EB}x_B + a_{EC}x_C + a_{ED}x_D\right) \\ &= \frac{1}{\lambda}\left(x_A + x_B\right) \end{aligned}$$

For the formula above, where λ is a constant and $A = [a_{ij}]$ is the adjacency matrix, we can express it in a matrix-vector notation:

$$\mathbf{x} = \frac{1}{\lambda} A\mathbf{x},$$

where

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_i \\ \vdots \\ x_n \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 0 & a_{12} & & a_{1j} & a_{1n} \\ a_{21} & 0 & & & a_{2n} \\ & & \ddots & & \\ a_{i1} & a_{i2} & & a_{ij} & a_{in} \\ a_{n1} & a_{n2} & & a_{nj} & 0 \end{bmatrix}$$

Following the friendship social network worked example, we have the system of equations below to describe the centrality scores for all the nodes, i.e.,

$$
\begin{aligned}
x_A &= \frac{1}{\lambda}\left(x_B + x_D + x_E\right) \\
x_B &= \frac{1}{\lambda}\left(x_A + x_C + x_D + x_E\right) \\
x_C &= \frac{1}{\lambda}\left(x_B + x_D\right) \\
x_D &= \frac{1}{\lambda}\left(x_A + x_B + x_C\right) \\
x_E &= \frac{1}{\lambda}\left(x_A + x_B\right)
\end{aligned}
$$

Five equations, Five unknowns!

This computation serves as a verification of the formula conversion.

We see that they are expressing the same system of equations in a slightly different manner:

$$\mathbf{x} = \frac{1}{\lambda} A\mathbf{x} \qquad \longrightarrow \qquad \lambda \times \mathbf{x} = \lambda \times \frac{1}{\lambda} A\mathbf{x} \qquad \longrightarrow \qquad \lambda\mathbf{x} = A\mathbf{x}.$$

Now, we can just move the constant λ to the left side of the formula and obtain the standard form for eigenvalues and eigenvectors, where λ is an eigenvalue and $\mathbf{x}$ is an eigenvector of the adjacency matrix.

In other words, we can rewrite the above expression in the following manner:

$$(\lambda I_n - A)\,\mathbf{x} = \mathbf{0},$$

i.e.,

$$\mathbf{x} = \frac{1}{\lambda} A\mathbf{x},$$

where

$$\lambda \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_i \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} 0 & a_{12} & & a_{1j} & a_{1n} \\ a_{21} & 0 & & & a_{2n} \\ & & \ddots & & \\ a_{i1} & a_{i2} & & a_{ij} & a_{in} \\ a_{n1} & a_{n2} & & a_{nj} & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_i \\ \vdots \\ x_n \end{bmatrix}$$

$$\lambda \begin{bmatrix} 1 & 0 & & 0 & 0 \\ 0 & 1 & & & 0 \\ & & \ddots & & \\ 0 & 0 & & 1 & 0 \\ 0 & 0 & & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_i \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} 0 & a_{12} & & a_{1j} & a_{1n} \\ a_{21} & 0 & & & a_{2n} \\ & & \ddots & & \\ a_{i1} & a_{i2} & & a_{ij} & a_{in} \\ a_{n1} & a_{n2} & & a_{nj} & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_i \\ \vdots \\ x_n \end{bmatrix}$$

At this point, given any social network, we can derive its adjacency matrix and compute its eigenvalues and eigenvectors.

In general though, there will be more than one eigenvalue for which an eigenvector solution exists.

In the social network analysis, however, we have an additional requirement. All the entries in the eigenvector need to be positive, i.e., $\lambda > 0$. This implies that only the greatest eigenvalue results in the desired centrality measure.

The conclusion above directly follows the Perron-Frobenius theorem, which asserts that a real square matrix with positive entries has a unique largest real eigenvalue and the corresponding eigenvector has strictly positive components.

This largest eigenvalue is called the **principal eigenvalue** and the corresponding eigenvector is called the **principal eigenvector**.

At this point, following the standard procedure for computing eigenvalues and eigenvectors, we are ready to compute the eigenvector centrality for each actor within a given network graph.

We will continue to use the friendship social network as our example.

We first compute the greatest eigenvalue.

For space considerations, the steps for computing the determinant of the adjacency matrix and solving the characteristic polynomial using the computation technique of the determinant are omitted.

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

$$\begin{aligned} \det(\lambda I_5 - A) &= \det\left(\lambda \begin{bmatrix} \color{red}{1} & 0 & 0 & 0 & 0 \\ 0 & \color{red}{1} & 0 & 0 & 0 \\ 0 & 0 & \color{red}{1} & 0 & 0 \\ 0 & 0 & 0 & \color{red}{0} & 0 \\ 0 & 0 & 0 & 0 & \color{red}{1} \end{bmatrix} - \begin{bmatrix} \color{red}{0} & 1 & 0 & 1 & 1 \\ 1 & \color{red}{0} & 1 & 1 & 1 \\ 0 & 1 & \color{red}{0} & 1 & 0 \\ 1 & 1 & 1 & \color{red}{0} & 0 \\ 1 & 1 & 0 & 0 & \color{red}{0} \end{bmatrix} \right) \\ &= \begin{bmatrix} \lambda & -1 & 0 & -1 & -1 \\ 1 & \lambda & -1 & -1 & -1 \\ 0 & -1 & \lambda & -1 & 0 \\ -1 & -1 & -1 & \lambda & 0 \\ -1 & -1 & 0 & 0 & \lambda \end{bmatrix} \\ &= \lambda^5 - 7\lambda^3 - 6\lambda^2 + 3\lambda + 2. \end{aligned}$$

Plugging in the greatest eigenvalue, we can compute its corresponding eigenvectors, which gives the centrality score solution for this network graph.

The following MATLAB statements are used to solve the eigenvector centrality:

```
clc
close all; clear all
A = [0 1 0 1 1;
1 0 1 1 1;
0 1 0 1 0;
1 1 1 0 0;
1 1 0 0 0];
[V,D] = eig(A);
evalmax = max(diag(D));
evec = null(evalmax*eye(5,5)-A);
evecsal = (1/max(evec))*evec
```

$$\begin{bmatrix} x_A \\ x_B \\ x_C \\ x_D \\ x_E \end{bmatrix} \approx \begin{bmatrix} 0.8407 \\ 1.0000 \\ 0.6270 \\ 0.8407 \\ 0.6270 \end{bmatrix}$$

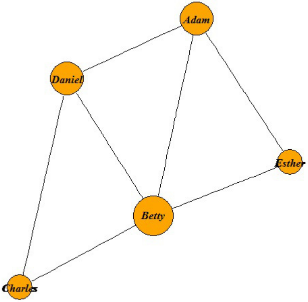
The eigenvector centrality is

$$\begin{bmatrix} C_{E,A} \\ C_{E,B} \\ C_{E,C} \\ C_{E,D} \\ C_{E,E} \end{bmatrix} \approx \begin{bmatrix} 0.8407 \\ 1.0000 \\ 0.6270 \\ 0.8407 \\ 0.6270 \end{bmatrix}$$

Eigenvector Centrality for a Friendship Social Network

Note that the eigenvector solution is not unique. Basically, we have found a set of eigenvectors in the following form, where α is a constant but the ratios are the same, which is what we care about:

$$\alpha \begin{bmatrix} 0.8407 \\ 1.0000 \\ 0.6270 \\ 0.8407 \\ 0.6270 \end{bmatrix}$$



We conclude that Betty is the most central person in a friendship social network.

Use R to check our results:

```r
# This loads the igraph package
library(igraph)
# choose an adjacency matrix from a .csv file
# e.g., trust.csv
dat=read.csv(file.choose(),header=TRUE,row.names=1,check.names=FALSE)
# coerces the data set as a matrix
m=as.matrix(dat)
m
# this will create an ’igraph object’
g=graph.adjacency(m,mode="undirected",weighted=NULL)
g
# Calculate the eigenvector centrality
eigen_centrality(g,directed=FALSE,weights=E(g)$weight)$vector
```

Use Python to check our results:

```
# Import required Python libraries (equivalent
    to R's igraph)
import pandas as pd
import networkx as nx

# Step 1: Read adjacency matrix from CSV file (
    equivalent to read.csv(file.choose()))
# In Python, we use a file dialog to select CSV
    (same interactive selection as R)
import tkinter as tk
from tkinter import filedialog
root = tk.Tk()
root.withdraw()  # Hide main window
file_path = filedialog.askopenfilename(filetypes
    =[("CSV Files", "*.csv")])
```

```
# Read CSV with row names as index, no header check (matches R)
dat = pd.read_csv(file_path, header=0, index_col=0)

# Step 2: Convert data to matrix (equivalent to as.matrix(dat))
m = dat.values
print("Adjacency Matrix:")
print(m)

# Step 3: Create undirected graph from adjacency matrix (NO weights, matches R)
g = nx.from_numpy_array(m, create_using=nx.Graph)
print("\nGraph Object Info:")
print(g)
```

```
# Step 4: Calculate eigenvector centrality (undirected, no weights)
# EXACT match to R's eigen_centrality(g, directed=FALSE, weights=NULL)$vector
eigen_centrality = nx.eigenvector_centrality(g, max_iter=1000)
print("\nEigenvector Centrality:")
for node, cent in eigen_centrality.items():
    print(f"Node {node}: {cent:.6f}")
```

It is suggested that the eigenvector of the largest eigenvalue of an adjacency matrix could make a good network centrality measure.

Unlike degree centrality, which weights every contact equally, the eigenvector weights contacts according to their own centrality.

Eigenvector centrality can also be seen as a weighted sum of direct and indirect connections of every length. Thus it takes into account the entire pattern of the network.

For example, in a trust social network, if a person is directly or indirectly trusted by other highly-trusted individuals, then this person tends to be trustworthy.

Star Graph

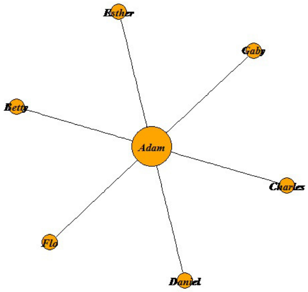


Figure 2: Eigenvector Centrality for a Star Graph

Circle Graph

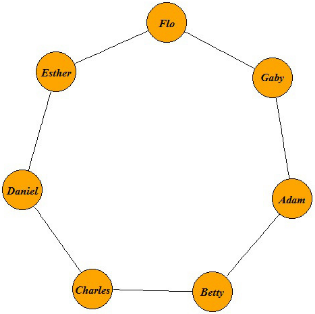


Figure 3: Eigenvector Centrality for a Circle Graph

Line Graph

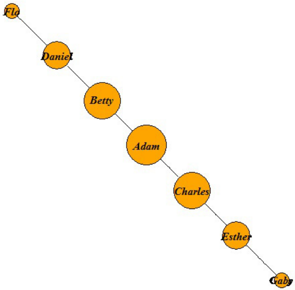


Figure 4: Eigenvector Centrality for a Line Graph

Bow Graph

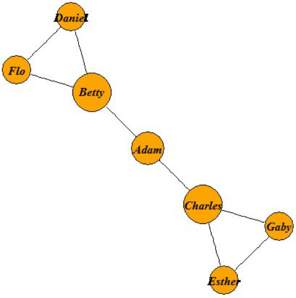


Figure 5: Eigenvector Centrality for a Bow Graph