Undirected Spectral Clustering Examples

Carlyle Morgan

5/24/2021

```
library(igraph)
```

```
## Warning: package 'igraph' was built under R version 4.0.5
```

```
library(dplyr)
library(stringr)
library(readr)
library(stats)
library(expm)
```

```
## Warning: package 'expm' was built under R version 4.0.5
```

Introduction

This dataset illustrates collaboration efforts between various academics in undirected network form. There are 86 distinct authors and 124 edges, each weighted by the number of collaborations between each of the authors.

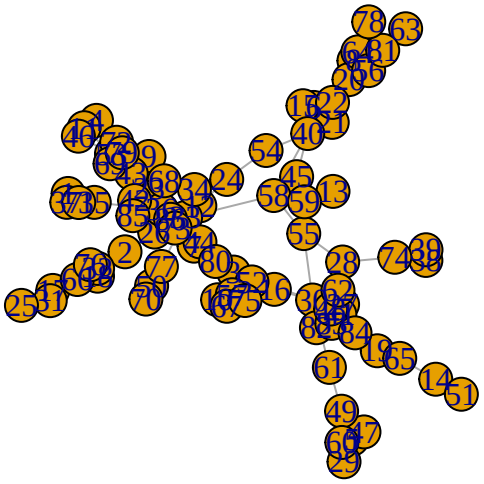
Preprocessing

Importing the dataset and reordering the to and from columns:

```
authors <- read_table2("C:/Users/sc_mo/Documents/R/Projects/Stats Research/ca-sandi_auths.csv")
auth_temp<-authors$node1
authors$node1<-authors$node2
authors$node2<-auth_temp
authlist<- unique(c(authors$node1, authors$node2))
```

Calling the network data into igraph and applying the edge weights as provided by the dataset:

```
g.authors<- graph.data.frame(authors[1:2], vertices = authlist, directed = FALSE)
E(g.authors)$weight<-authors$weight
set.seed(123)
plot(g.authors)
```



Checking for connected components and simplicity:

```
is.simple(g.authors)
```

```
## [1] TRUE
```

```
is.connected(g.authors)
```

```
## [1] TRUE
```

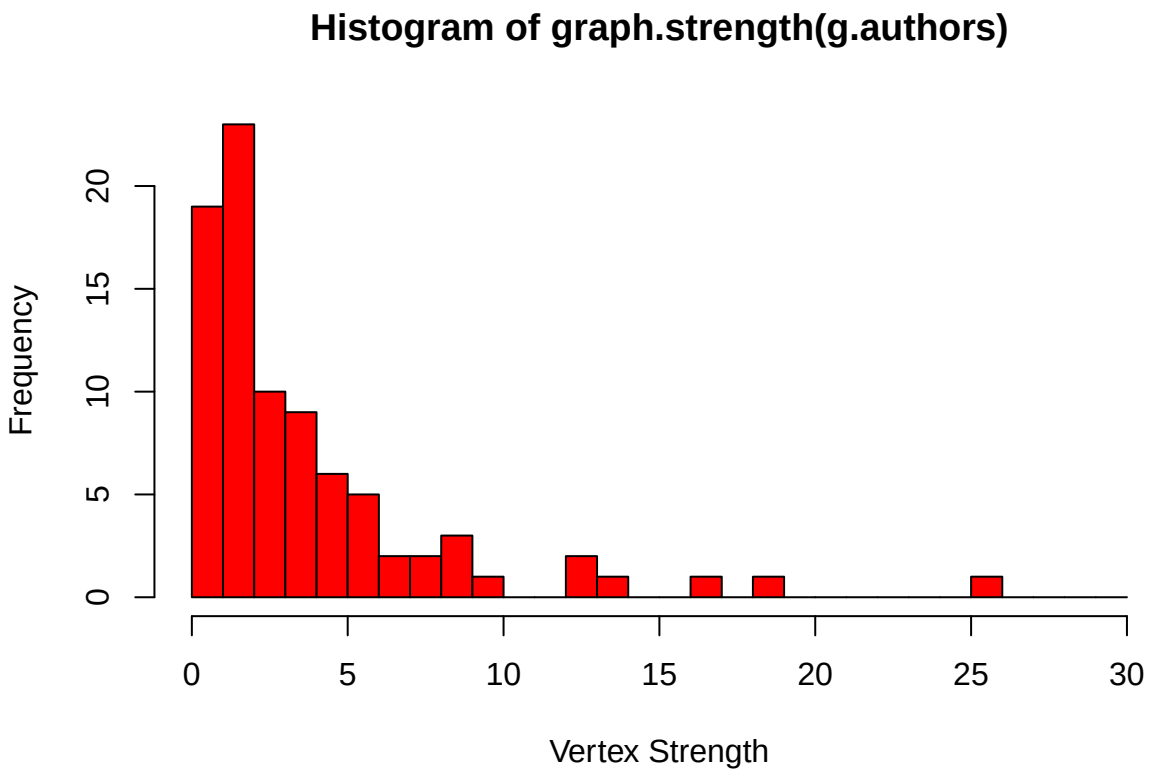
```
head(sort(graph.strength(g.authors), decreasing=TRUE),25)
```

```
## 36 33 43 53  3 30 44 20 23 35 40 66  2 58 26 49 55 60 59 17 45 64 83 84 79
## 26 19 17 14 13 13 10  9  9  9  8  8  7  7  6  6  6  6  6  5  5  5  5  5  5
```

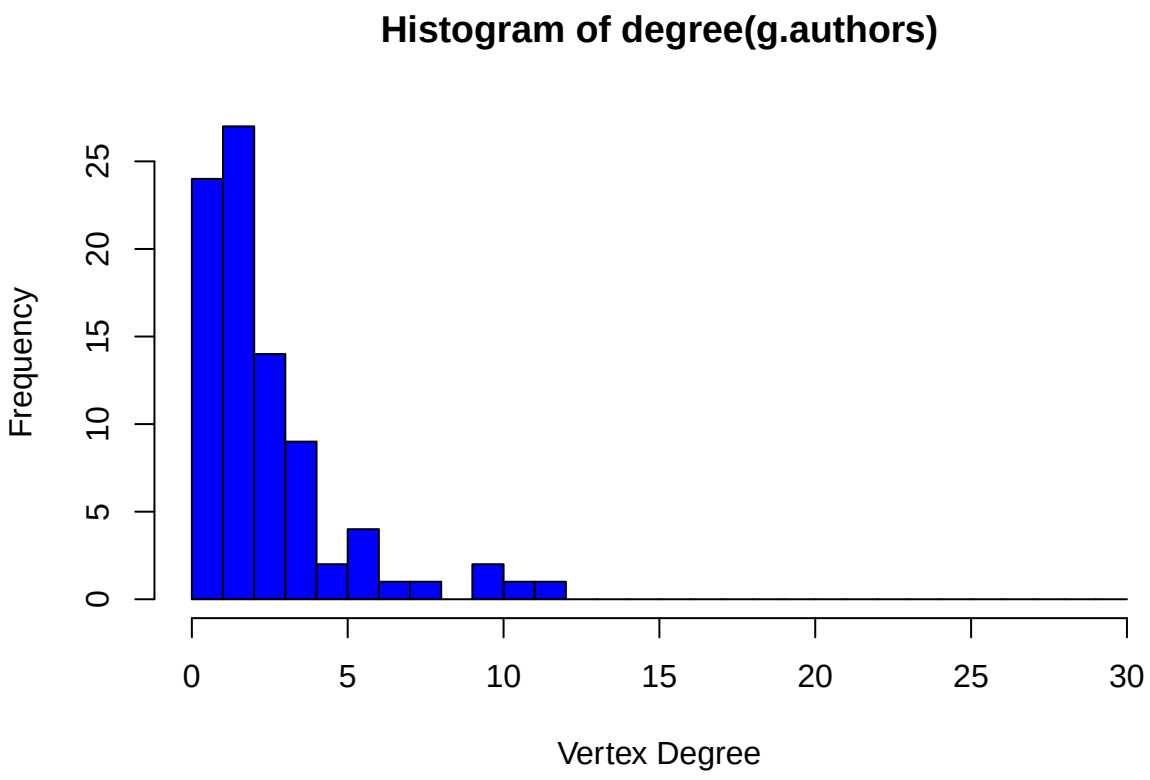
```
head(sort(degree(g.authors), decreasing=TRUE),25)
```

```
## 30 36  3 33 40 53  2 20 43 66 44 58 18 26 32 35 49 55 76 59 84 17 23 27 28
## 12 11 10 10  8  7  6  6  6  6  5  5  4  4  4  4  4  4  4  4  4  3  3  3  3
```

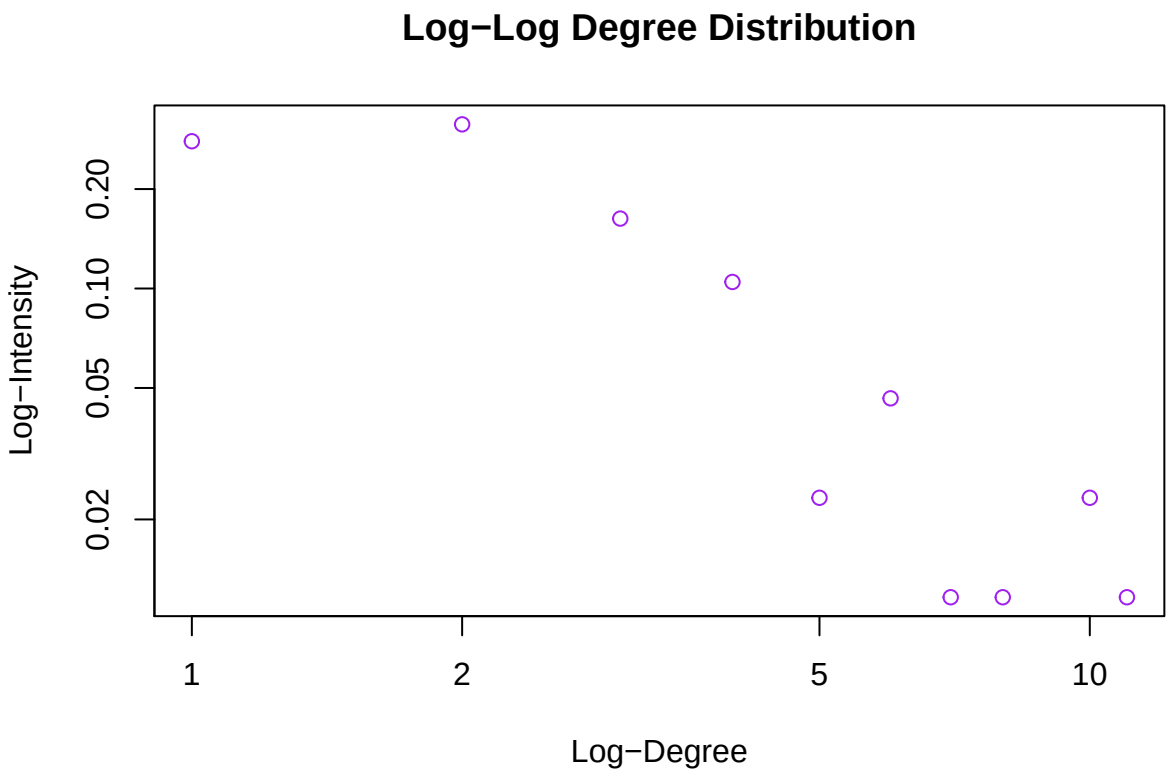
```
hist(graph.strength(g.authors), col="red", xlab="Vertex Strength",
     ylab="Frequency",
     xlim = c(0,30), breaks = seq(0,30,by=1))
```




```
hist(degree(g.authors), col="blue", xlab="Vertex Degree",
     ylab="Frequency", xlim = c(0,30), breaks = seq(0,30,by=1))
```




```
d.rh <- degree(g.authors)
dd.rh <- degree.distribution(g.authors)
d <- 1:max(d.rh)-1
ind <- (dd.rh != 0)
plot(d[ind], dd.rh[ind], log="xy", col="purple", xlab=c("Log-Degree"), ylab=c("Log-Intensity"), main="L
```



Creating the Necessary Laplacian Components

Creating the weighted(W) and unweighted(A) adjacency matrices:

```
A<-get.adjacency(g.authors)
A<-as.matrix(A)
W<-A
for (i in 1:nrow(authors)){
  W[as.character(authors$node1[i]), as.character(authors$node2[i])]<-authors$weight[i]
  W[as.character(authors$node2[i]), as.character(authors$node1[i])]<-authors$weight[i]
}
```

Creating the degree matrix(D):

```
D<-diag(strength(g.authors))
```

Generating the Laplacian Matrices:

Creating the unnormalized Laplacian Matrix, $L_{un} = D - W$:

```
L_unnorm<-D-W
```

Creating the normalized symmetric Laplacian Matrix, $L_{sym} = D^{-1/2} L_{un} D^{-1/2}$:

```
D_neghalf<-1/sqrt(D)
D_neghalf[D_neghalf==Inf]<-0
L_sym<-D_neghalf %*% L_unnorm %*% D_neghalf
```

Creating the normalized random walk Laplacian Matrix, $ L_{rw} = D^{-1}L_{un}$

```
D_negone<-1/D
D_negone[D_negone==Inf]<-0
L_rw<-D_negone %*% L_unnorm
```

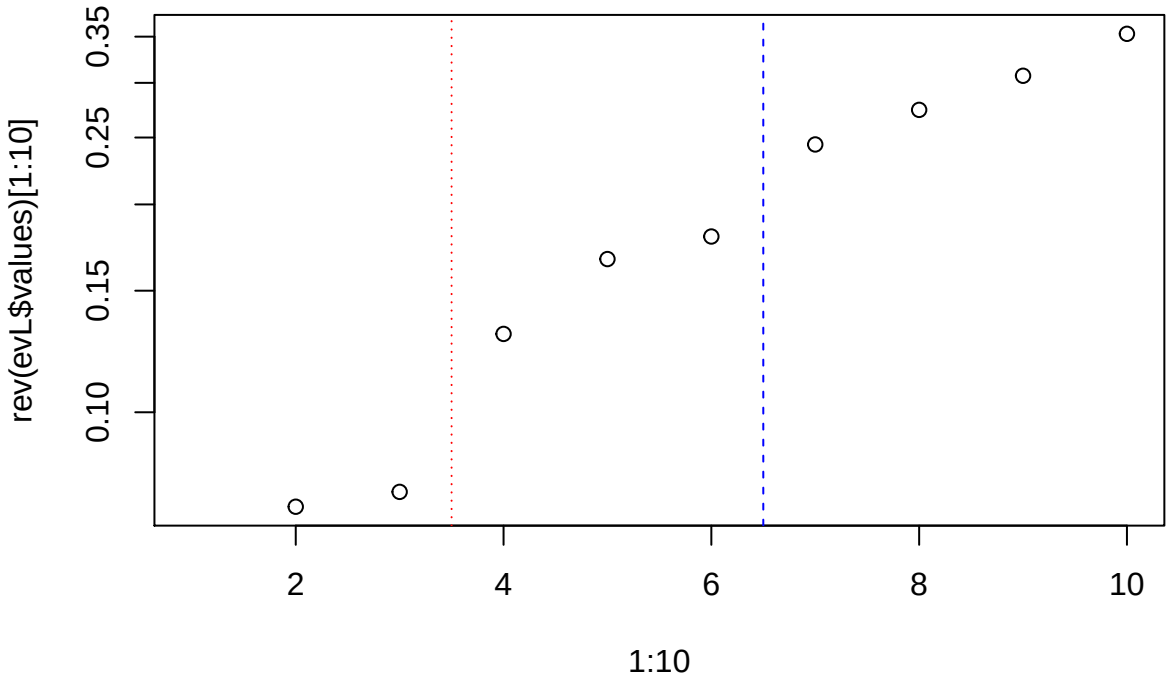
Unnormalized Spectral Clustering

Calculating the eigenvectors of L_{un}:

```
evL <- eigen(L_unnorm, symmetric=TRUE)
```

Creating a simple plot of the eigenvalues:

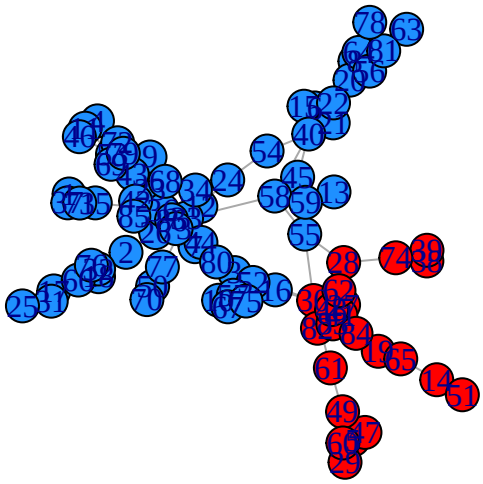
```
plot(1:10,rev(evL$values)[1:10], log="y")
abline(v=3.5, col="red", lty=3)
abline(v=6.5, col="blue", lty=2)
```



Clustering with k=2 clusters:

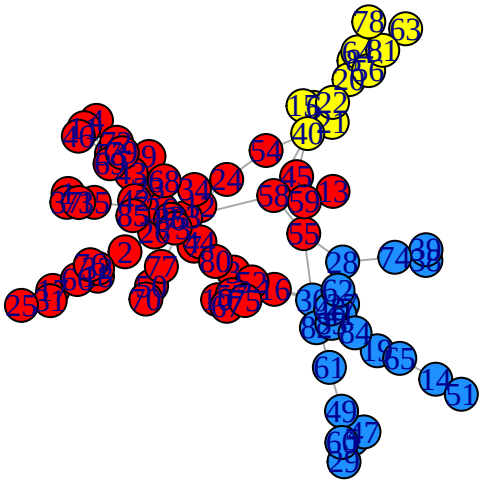
```
k   <- 2
Z   <- evL$vectors[,(ncol(evL$vectors)-k+1):ncol(evL$vectors)]

km <- kmeans(Z, centers=k, nstart=5)
V(g.authors)$cluster<-km$cluster
V(g.authors)[cluster == 1]$color<-"red"
V(g.authors)[cluster == 2]$color<-"dodgerblue"
set.seed(123)
```

```
plot(g.authors)
```



Clustering with k=3 clusters:

```
k   <- 3
Z   <- evL$vectors[,(ncol(evL$vectors)-k+1):ncol(evL$vectors)]

km <- kmeans(Z, centers=k, nstart=5)
V(g.authors)$cluster<-km$cluster
V(g.authors)[cluster == 1]$color<-"red"
V(g.authors)[cluster == 2]$color<-"dodgerblue"
V(g.authors)[cluster == 3]$color<-"yellow"
set.seed(123)
plot(g.authors)
```



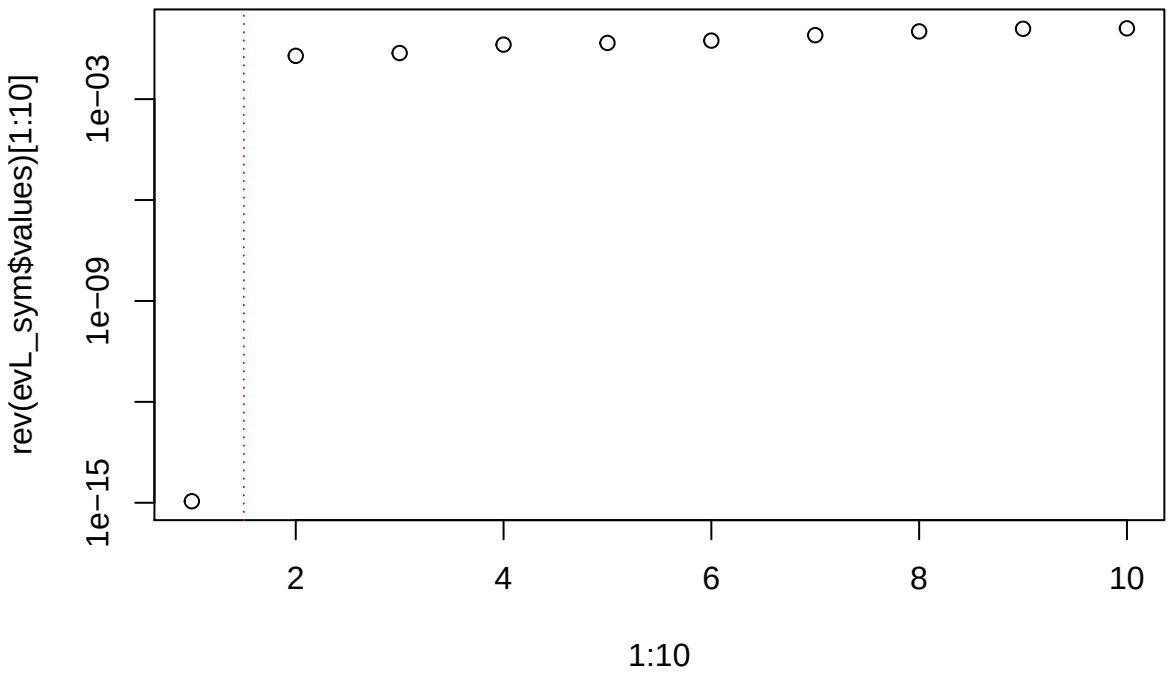
Normalized Spectral Clustering with the Symmetric graph Laplacian

Calculating the eigenvectors of L_{sym}:

```
evL_sym <- eigen(L_sym, symmetric=TRUE)
```

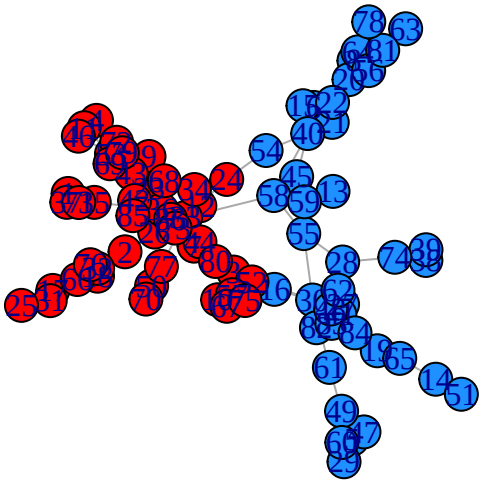
Creating a simple plot of the eigenvalues:

```
plot(1:10,rev(evL_sym$values)[1:10], log="y")
abline(v=1.5, col="red", lty=3)
```



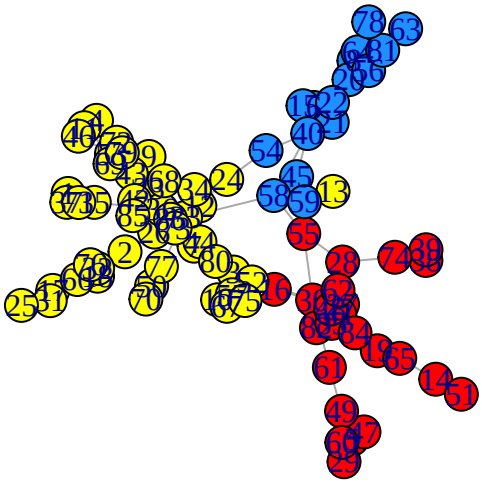
Clustering with k=2 clusters:

```
k   <- 2
Z   <- evL_sym$vectors[,(ncol(evL_sym$vectors)-k+1):ncol(evL_sym$vectors)]


km <- kmeans(Z, centers=k, nstart=5)
V(g.authors)$cluster<-km$cluster
V(g.authors)[cluster == 1]$color<-"red"
V(g.authors)[cluster == 2]$color<-"dodgerblue"
set.seed(123)
plot(g.authors)
```



Clustering with k=3 clusters:

```
k   <- 3
Z   <- evL_sym$vectors[,(ncol(evL_sym$vectors)-k+1):ncol(evL_sym$vectors)]

km <- kmeans(Z, centers=k, nstart=5)
V(g.authors)$cluster<-km$cluster
V(g.authors)[cluster == 1]$color<-"red"
V(g.authors)[cluster == 2]$color<-"dodgerblue"
V(g.authors)[cluster == 3]$color<-"yellow"
set.seed(123)
plot(g.authors)
```



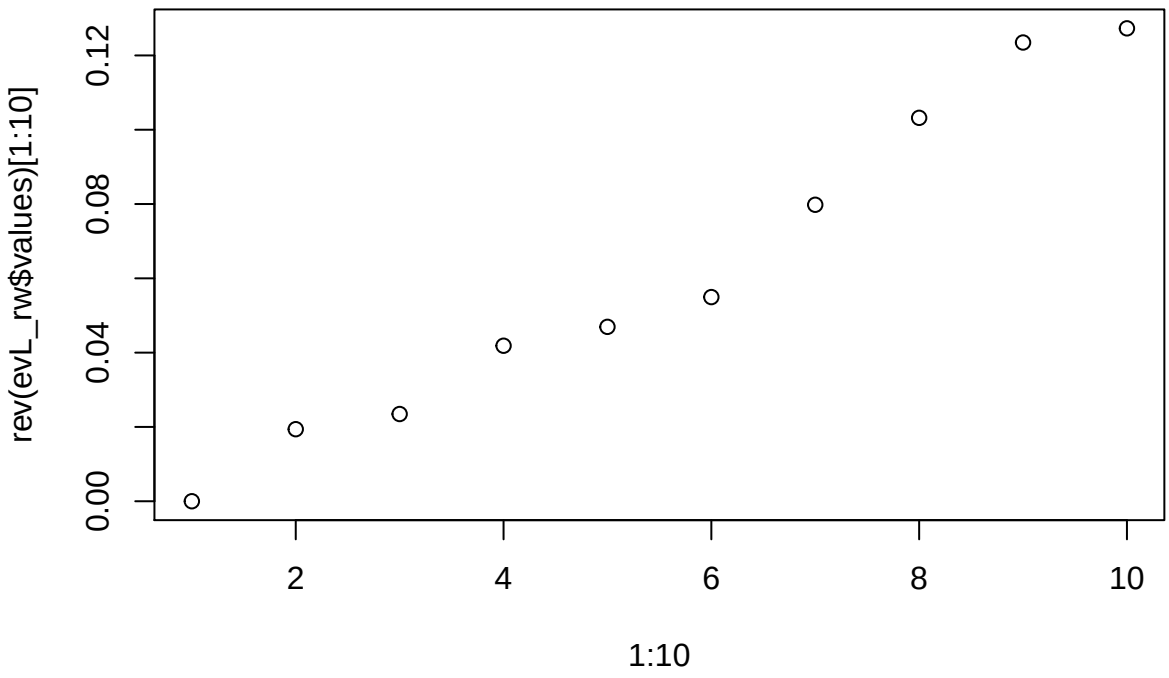
Normalized Spectral Clustering with the Random Walk graph Laplacian

Calculating the eigenvectors of L_{sym}:

```
evL_rw <- eigen(L_rw, symmetric=FALSE)
```

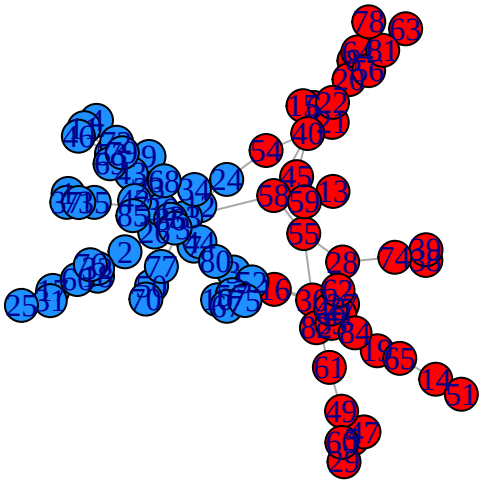
Creating a simple plot of the eigenvalues:

```
plot(1:10,rev(evL_rw$values)[1:10])
```



Clustering with k=2 clusters:

```
k   <- 2
Z   <- evL_rw$vectors[,(ncol(evL_rw$vectors)-k+1):ncol(evL_rw$vectors)]

km <- kmeans(Z, centers=k, nstart=5)
V(g.authors)$cluster<-km$cluster
V(g.authors)[cluster == 1]$color<-"red"
V(g.authors)[cluster == 2]$color<-"dodgerblue"
set.seed(123)
plot(g.authors)
```



Clustering with k=3 clusters:

```
k   <- 3
Z   <- evL_rw$vectors[,(ncol(evL_rw$vectors)-k+1):ncol(evL_rw$vectors)]

km <- kmeans(Z, centers=k, nstart=5)
V(g.authors)$cluster<-km$cluster
V(g.authors)[cluster == 1]$color<-"red"
V(g.authors)[cluster == 2]$color<-"dodgerblue"
V(g.authors)[cluster == 3]$color<-"yellow"
set.seed(123)
plot(g.authors)
```