Key Actor Analysis in R
Integrating SNA and Statistics*

Manuel S. González Canché
msgc@email.arizona.edu
University of Arizona

Cecilia Rios Aguilar
Cecilia.Rios-Aguilar@cgu.edu
Claremont Graduate University

INCHER, University of Kassel

June 25, 2012

Contents

*We are thankful with Drew Conway (New York University - Department of Politics) for all the help provided to conduct these analyses.

1 Presentation

The purpose of this document is to explain the process followed to identify actors that are playing important roles in a given network. The ultimate goal is to gather as much information as possible to understand the network and identify players that could influence the behavior of the group. With this understanding, by identifying and reaching key players, and given the dynamic nature of social organizations, researchers could be in better positions to interact more effectively with the group.

Specifically, we will focus on two measures of centrality: eigenvector and betweenness centralities, yet the procedures presented in the tutorial could be easily extended to other centrality measures as it fits the needs of a specific project.

The analysis will be conducted in the R platform. The advantage of using R in this case is the easiness to conducting econometric or statistical methods using centrality measures.

This document is divided in three main sections, the first corresponds brief recapitulation of the centrality measures and the data used; the second corresponds to the steps followed to plot eigenvector and betweenness centralities weighed by their residuals, several plot options will be offered. Finally, we will present the sociogram that shows the relevance of using these two measures to understand the network and identify key actors.

Note: This document is for reference only, please do not copy any command from here. The code contained is in R language and requires "perfection." Copying and pasting from this document is likely to contain noise and consequently the functions won't work, or won't work properly. Instead use the text document called "Key Actor Analysis.R" provided during the session which will be described later in this tutorial.

2 Contextualization

Purpose

- SNA is often used to identify central or key actors within a social group
- Given the **dynamic nature** of social groups the identification of key players may be critical in any attempt to influence the behavior of the network

  Purpose
  At the end of the session the participants are expected to understand and replicate the procedures followed to conduct key actor analysis in R.

  We will use two centrality measures and linear regression procedures to conduct the analysis.

The dataset comes from real data extracted from a Facebook application that creates virtual communities within community colleges across the United States of America. Each community college invites students to form part of this virtual community and we have requested authorizations to students to use this dataset. All identifiable information has been removed.

2.1 Why R?

In this case

- R will immediately allow us to compute and save the centrality measures that we later will use as an input in the regression model fitted.
- The graphical capacities of R also represent a series of advantages and functions that is are not available in other softwares.

3 Review of degree measures

Quick review of centrality measures

To identify key actor SNA relies on:

Degree:	Number of connections an actor has
Betweenness	Number of shortest paths an actor is on which makes this actor important in controlling the flow of information in the network.
Closeness	Relative distance of one actor to all other actors
Eigenvector	A measure of how central an actor is and how central the ties of this actors are in the network.

- **Methodologically**, all these measures come from the same matrices and to some extent share **mathematical properties**
- Consequently, these measures are expected to have linear relationships

3.1 Building upon linear relationships of centrality measures

Plotting centrality measures...

- A method for using centrality metrics to identify key actors is to plot actors' scores for Eigenvector centrality versus Betweenness
- Although this measures are correlated, they are not perfectly linear
- In this sense, we can use these non-linear outliers to enrich our knowledge of the social relationships in the network

Conceptual implications

1. An actor with very high betweenness but low EC may be a critical gatekeeper to a central actor
2. Likewise, an actor with low betweenness but high EC may have unique access to central actors

What we have just said translates into the following:

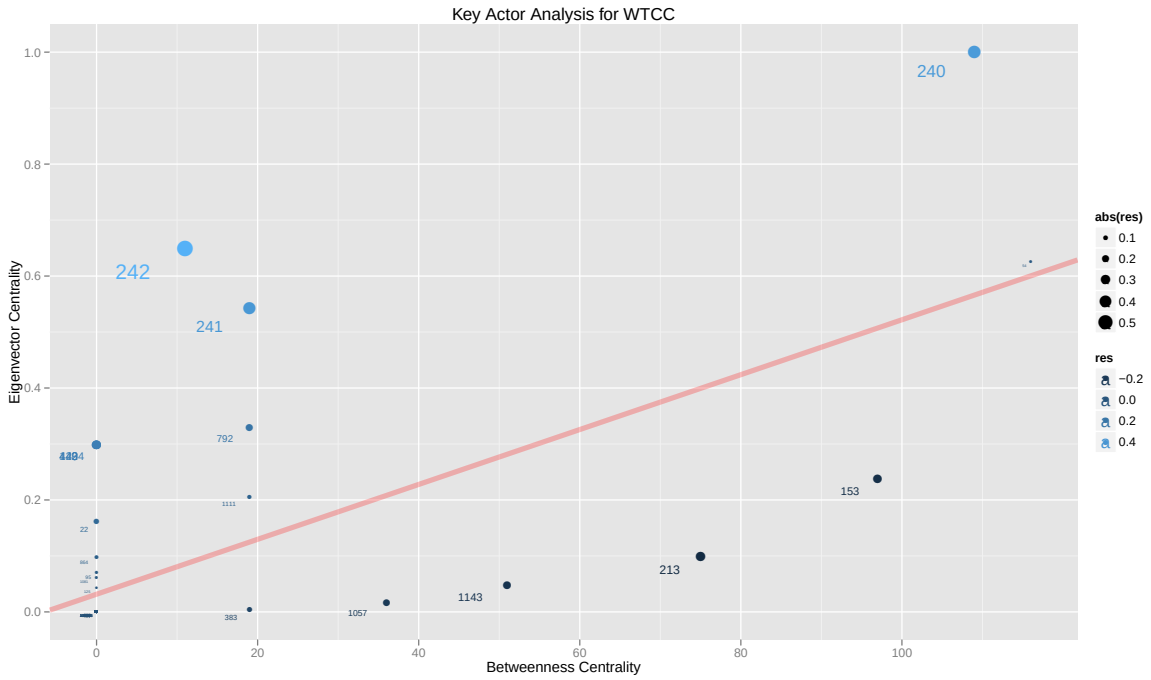

Figure 1: Key Actor analysis, this is part of what we will replicate today!

4 More traditional, yet novel SNA approach

Using the centrality measures to highlight key actors

- Using our network data we will identify the location of the key actors from the previous analysis
- Theoretically, our interpretation should completely be congruent to what was found in the bivariate plot
- We can use the regression residuals if want to, or simply use eigenvector or betweenness centralities as attributes and conditional on what we want the sociogram to show.

Figure 2 represents a sociogram with default options, which very few times is as informative as it should be. Our goal is to obtain a sociogram like the Figure 3 on page 6.

5 R functions

In this section we present the R codes to replicate Figures 1 and 3.

5.1 Code to replicate Figure 1

Depending on where you have placed the dataset the path will need to be modified accordingly. Also, Mac and Pc will have small differences in terms of how to write

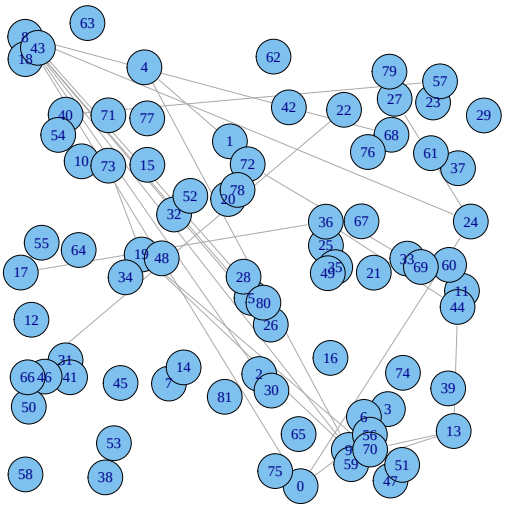

Figure 2: As usual, the default option is not revealing much!

paths, for instance:

Possible paths:

```
#Sets workding directory
setwd("C:/Users/MSGC/Documents/Ph. D/Dropbox/Germany PHUDCFILY
/TricDataWorkshop") #Pc

setwd("/Users/msgc/Dropbox/Dropbox/Germany PHUDCFILY
/TricDataWorkshop") #mac
```

Please follow the next directions:

1. Open R.
2. Remember to modify the path based on your computer's configuration.
3. Copy and paste one of the previous paths (depending on whether you are using a mac or Pc).
4. Once you are sure about having correctly modified the path, paste it in R and hit enter. This command told R where to read the data.

The following command will import our previously modified *.csv into the R environment. Remember, this dataset needs to be located in the folder indicated by our previous path.

We start by getting the data ready

1. Just in case, setting working directory
   `setwd("C:/Users/MSGC/Documents/Ph. D/Dropbox/Germany PHUDCFILY /TricDataWorkshop") #Pc`

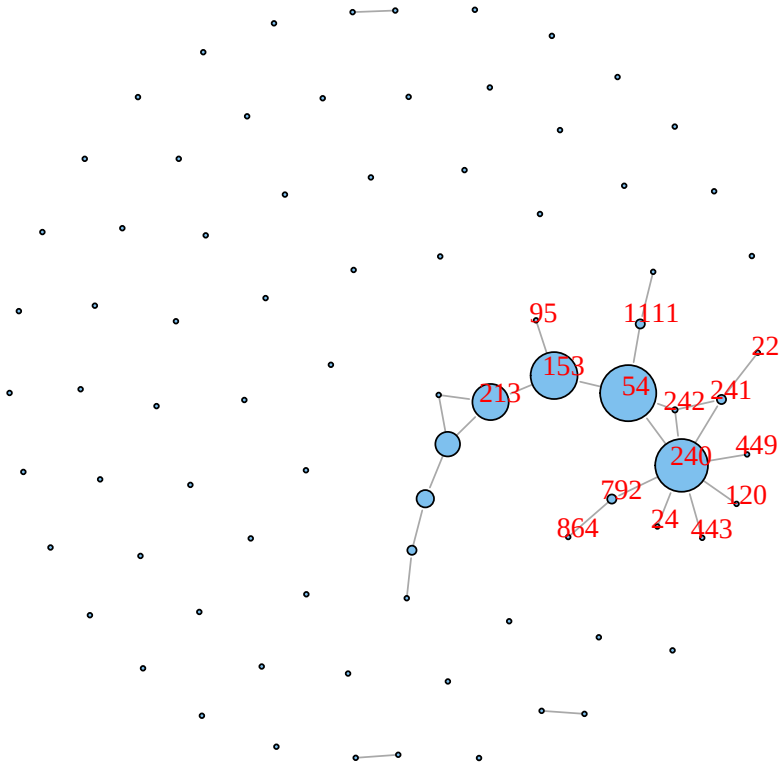

Figure 3: Key actors weighed by eigenvector and betweenness centrality

2. To read the data from csv format located in our working directory
```
dta <- read.csv("User_to_User_Friendships.csv", header = TRUE,
sep = ",", skip = 0,row.names = NULL)
```

3. Add row names based on the information contained in column 1
```
rownames(dta)<-dta[,1]
```

4. Tells us dimensions (No. of rows and columns), which in this case is problematic because R would not read a non square matrix when we want to analyze a one mode network
```
dim(dta)
```

5. Removes first column to get squared matrix, but I want to keep the original data (dta), just in case, so I save it as dta2
```
dta2<-dta[,-1]
```

6. Making sure it worked `dim(dta2)`

7. Shows the data
   `fix(dta)`

8. Loading our package, due to very recent updates, in case this coding does not work, we can install the previous version of igraph, called igraph0, using this command install.packages("igraph0")
   `library(igraph)`

9. Getting our graph object
   `G<-graph.adjacency(dta2, mode=c("undirected"))`

10. Creating a dataset that contains the two measures of interest in this case. It is worth mentioning that evcent returns lots of data associated with the, but we only need the leading eigenvector
    `cent<-data.frame(bet=betweenness(G),eig=evcent(G)$vector)`

11. Adding rownames to the dataset cent using our previously saved rownames from dta
    `rownames(cent)<-rownames(dta)`

12. We now regress eigenvector on betweenness, but just saving the residuals from the regression
    `res<-lm(eig~bet,data=cent)$residuals`

13. We modified the database called cent by adding a row with the residuals
    `cent<-transform(cent,res=res)`

14. Congratulations, we now have all we need to plot the data

Now we are ready to start plotting

1. In case we do not have it, we should install and load the package ggplot2 which is a very powerful but complicated platform, yet we use it to make things prettier.
   `install.packages("ggplot2")`
   `library(ggplot2)`

2. in the following code, cet is the dataset and we specifically are asking for two variables eig and bet, in addition we need the row names to be considered in the plot. Additionally, we are asking the bubbles or names to be weighed by the size of the residuals and colored by their absolute value as well.
   `p<-ggplot(cent,aes(x=bet,y=eig, label=rownames(cent),colour=res,`
   `size=abs(res)))+xlab("Betweenness Centrality")+ylab("Eigenvector Centrality")`

3. p is the graph all the information necessary to, yet it requires some extra commands called adding layers, for instance geom_point() will add bubbles
   `p+geom_point()+opts(title="Key Actor Analysis for WTCC")`

4. geom_text() adds names instead
   `p+geom_text()+opts(title="Key Actor Analysis for WTCC")`

5. We, of course can add both, just adjusting them so that they do not overlap
   ```
   p + geom_point() + geom_text(hjust=2, vjust=2)+
   opts(title="Key Actor Analysis for WTCC")
   ```

6. We can add the best line possible based on the data, to do this we need to compute the linear regression one more time and save the α and β coefficients
   `coeffs<-as.data.frame(coef(lm(eig~bet,data=cent)))`

7. Then we just simply add the new variables to our old graph
   ```
   p + geom_point() + geom_text(hjust=2, vjust=2)+opts(title="Key Actor
   Analysis for WTCC") + geom_abline(intercept = coeffs[1,], slope =
    coeffs[2,],colour = "red", size = 2,alpha=.25)
   ```

5.2 Code to replicate Figure 3

1. Lets initialize igraph
   `library(igraph)`

2. Getting our graph object (which we already had from Figure 1)
   `G<-graph.adjacency(dta2, mode=c("undirected"))`

3. Setting up the algorithm that provides a layout that allows for the appreciation of centrality measures
   `l<-layout.fruchterman.reingold(G, niter=10000)`

4. Adding the names of our database dta to the network
   `V(G)$name<-rownames(dta)`

5. Adding a size to each not that is proportional to its betweenness centrality, where the the divisor is the highest betweenness found in the network.
   `V(G)$size<-abs((cent$bet)/116)*15`

6. In this step we are setting a variable that allows us to manipulate the names of the nodes that will be shown in the final graph `nodes<-V(G)$name`

7. We can then select the centrality measure that better highlights the key actors, in this case we are using eigenvector centrality.
   `nodes[which(abs(cent$eig)<.064)]<-NA`

8. We are ready to plot or draw the map!
   ```
   plot(G,layout=l,vertex.label=nodes, vertex.label.dist=0.25,
   vertex.label.color="red",edge.width=1)
   ```

9. We can also automatize the process and ask R to save a pdf
```
pdf("actor_plot.pdf")
plot(G,layout=l,vertex.label=nodes, vertex.label.dist=0.25,
vertex.label.color="red",edge.width=1)
title(main="Kek Actor Analysis", sub="Key actors weighed by eigenvector
and betweenness centrality", col.main="black", col.sub="black",
cex.sub=1.2,cex.main=2,font.sub=2) dev.off()
```

Enjoy!