SMI 606: Discovery Using Social Network Analysis

Dr. Todd Hartman
Sheffield Methods Institute

Roadmap: Week 8

- Lecture: Social network analysis (SNA)
- Exercise: Finding network data
- Lab: Analyzing social network data using Facebook

Metadata

Network data is *metadata*

> *"metadata absolutely tells you everything about somebody's life. If you have enough metadata, you don't really need content" Stewart Baker*

> *"We kill people based on metadata" Michael Hayden*

Metadata

Network data is metadata

- A called B but there is no transcript of the conversation
- A and B both know C
- B met C in A's house but we don't know what they did

Connections

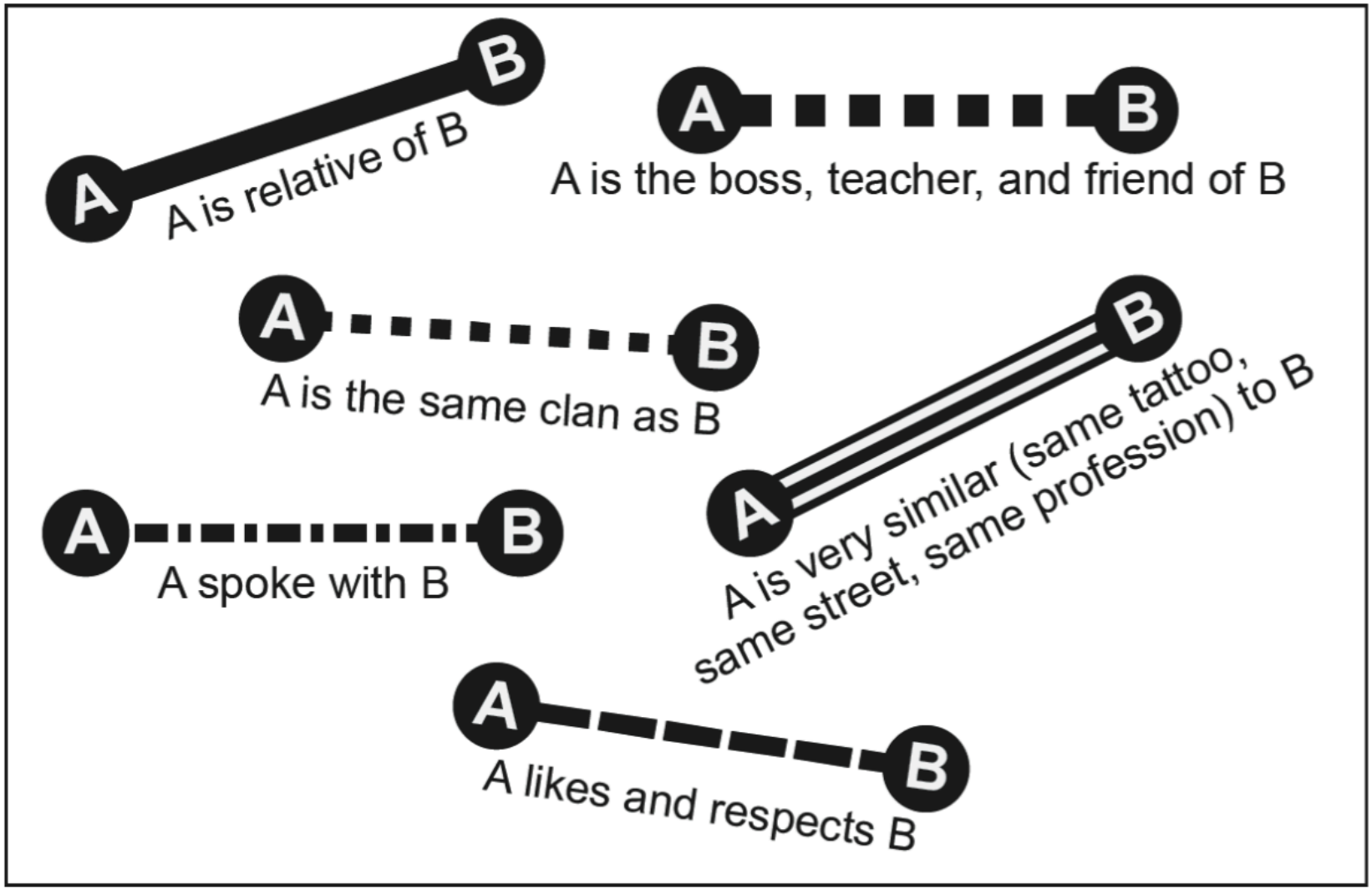


Figure: Edges, from the field manual

Network terminology

A and B are nodes or **vertices**, connected by **edges** or ties.
We are often interested in identifying 'brokers' – individuals who are the connection between cohesive groups.

A famous broker

Who was ‘Abu Ahmed al-Kuwaiti’?

A famous broker

Who was ‘Abu Ahmed al-Kuwaiti’?
Al Quaida commanders A, B, and C meet with Kuwaiti
Kuwaiti meets brother, ‘The Pacer’, and others in an Abbottabad compound

A famous broker

Who was ‘Abu Ahmed al-Kuwaiti’?
Al Quaida commanders A, B, and C meet with Kuwaiti
Kuwaiti meets brother, ‘The Pacer’, and others in an Abbottabad compound
‘The Pacer’ is Osama bin Laden
Kuwaiti is the connection between the group at Abbottabad and groups in Afghanistan

Social network analysis

Example: Data from a large insurgency

- We'll use anonymised data: `anon.data`

Social network analysis

Example: Data from a large insurgency

- We'll use anonymised data: `anon.data`

Task:

- Identify the key players with measures of network centrality
- ...

Social network analysis

The insurgency organizes different sectors of the society in social groups, clubs, etc.

- **Vertices** are potential insurgents
- **Edges** occur when two potential insurgents are members of the same group (weighted by the *number of groups* they share membership of)

Social network analysis

Here, rows are insurgents and columns are groups

```
head(anon.data, 3)
```

```
   SAL LN NC LRC TP BC LE
JA   0  0  1   1  0  0  0
SA   0  0  1   1  0  1  1
DA   0  0  1   0  0  0  0
```

```
apn[1:3,1:3] ## transformed to network data
```

```
   JA SA DA
JA NA  2  1
SA  2 NA  1
DA  1  1 NA
```

How did we do that?

```
np <- nrow(anon.data)
insurgents <- rownames(anon.data)
apn <- matrix(NA, nrow=np, ncol=np)
rownames(apn) <- insurgents
colnames(apn) <- insurgents
for (i in 1:np){
 for (j in 1:i){
   if (i != j){
     share <- sum(anon.data[i,] * anon.data[j,])
     apn[i,j] <- share
     apn[j,i] <- share
   }
 }
}
```

How could we do that again?

```
make.adj <- function(dd){
  np <- nrow(dd)
  peeps <- rownames(dd)
  apn <- matrix(NA, nrow=np, ncol=np)
  rownames(apn) <- colnames(apn) <- peeps
  for (i in 1:np){
    for (j in 1:i){
      if (i != j){
        share <- sum(dd[i,] * dd[j,])
        apn[i,j] <- apn[j,i] <- share
      }
    }
  }
  return(apn)
}
```

Social network analysis

Make a graph from it

```
apn.graph <- graph.adjacency(apn,
                mode="undirected", diag=FALSE)
```

Like the marriage network and *unlike* the Twitter follower network

- this is a **undirected** network (that's why the adjacency matrix was symmetric)

We'll consider three different ways to understand **network centrality**

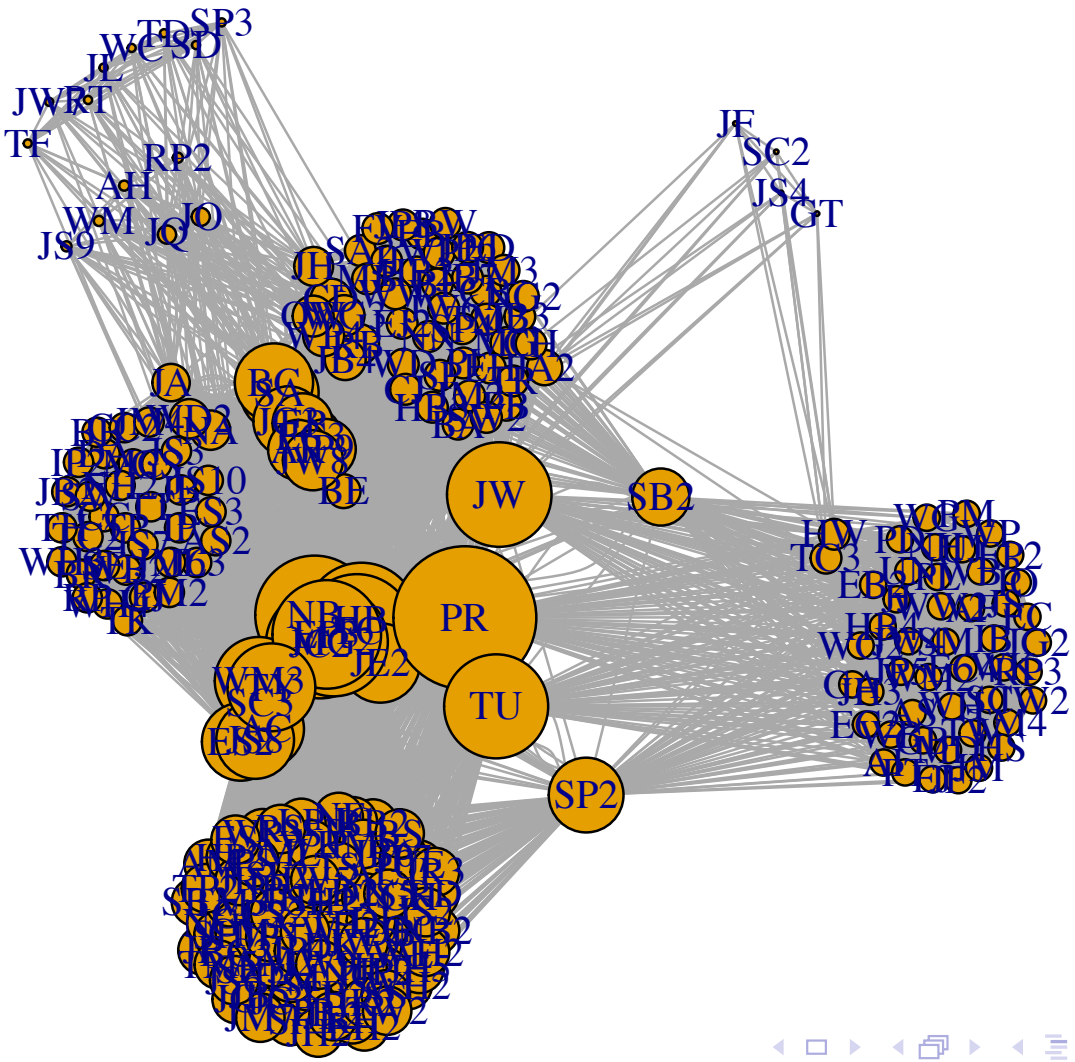
Local centrality measures

Degree: How many connections does each node have?

```
deg <- degree(apn.graph)
summary(deg)

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   9.00   53.50   61.00   78.95   96.00  283.00
```

Degree visualised



Suspiciously close...

Who has the largest numbers of connections?

```
degree.suspects <- sort(deg, decreasing=TRUE)
degree.suspects[1:10]

 PR  NB  HB  TC JC2 MG2 EP3  JW  TU WM3
283 235 224 224 215 215 215 207 206 174
```

Global centrality measures: distance

For an organiser it's important to be able to reach people easily.
Maybe we should look for nodes *close* to lots of others
The **distance** between A and B is the *number of edges* in the *shortest path* between them
On *average* this is

```
mean(distances(apn.graph))

[1] 1.692975
```

Global centrality measures: closeness

farness: sum of distances between a node and every other.

```
dist <- distances(apn.graph)
```

Consider SA.

```
sum(dist['SA',])

[1] 387
```

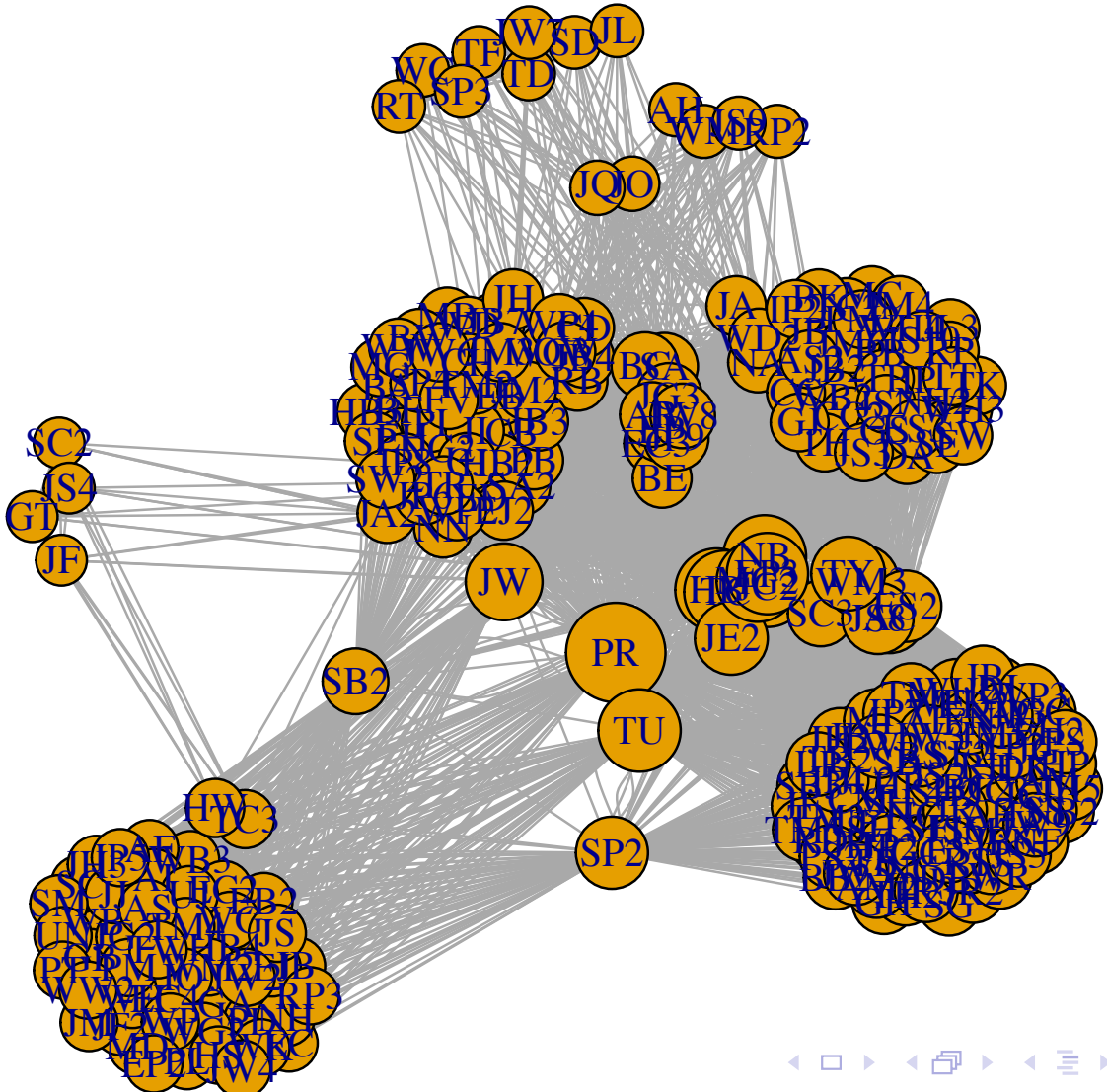
the inverse of farness is **closeness**

```
1/sum(dist['SA',])

[1] 0.002583979
```

(He *averages* 1.523622 links)

Closeness visualised



Global centrality measures: closeness

Fortunately, we have a convenient function

```
closeness(apn.graph)['SA'] # 0.002583979
```

Let's use this measure to see who we might want to talk to...

```
closeness.suspects <- sort(closeness(apn.graph), decreasing=TRUE)
closeness.suspects[1:10]
```

```
         PR          NB          HB          TC          TU
0.003831418 0.003194888 0.003194888 0.003194888 0.003174603
        JC2         MG2         EP3          JW         WM3
0.003134796 0.003134796 0.003134796 0.002958580 0.002840909
```

Global centrality measures: betweenness

We have used ‘shortest path’ to operationalize ‘distance’ in a network
and summed over the lengths of these paths to measure closeness
But let’s consider the paths themselves. . .

Global centrality measures: betweenness

We have used ‘shortest path’ to operationalize ‘distance’ in a network
and summed over the lengths of these paths to measure closeness
But let’s consider the paths themselves. . .
How to find nodes that are on *many* shortest paths?

Global centrality measures: betweenness

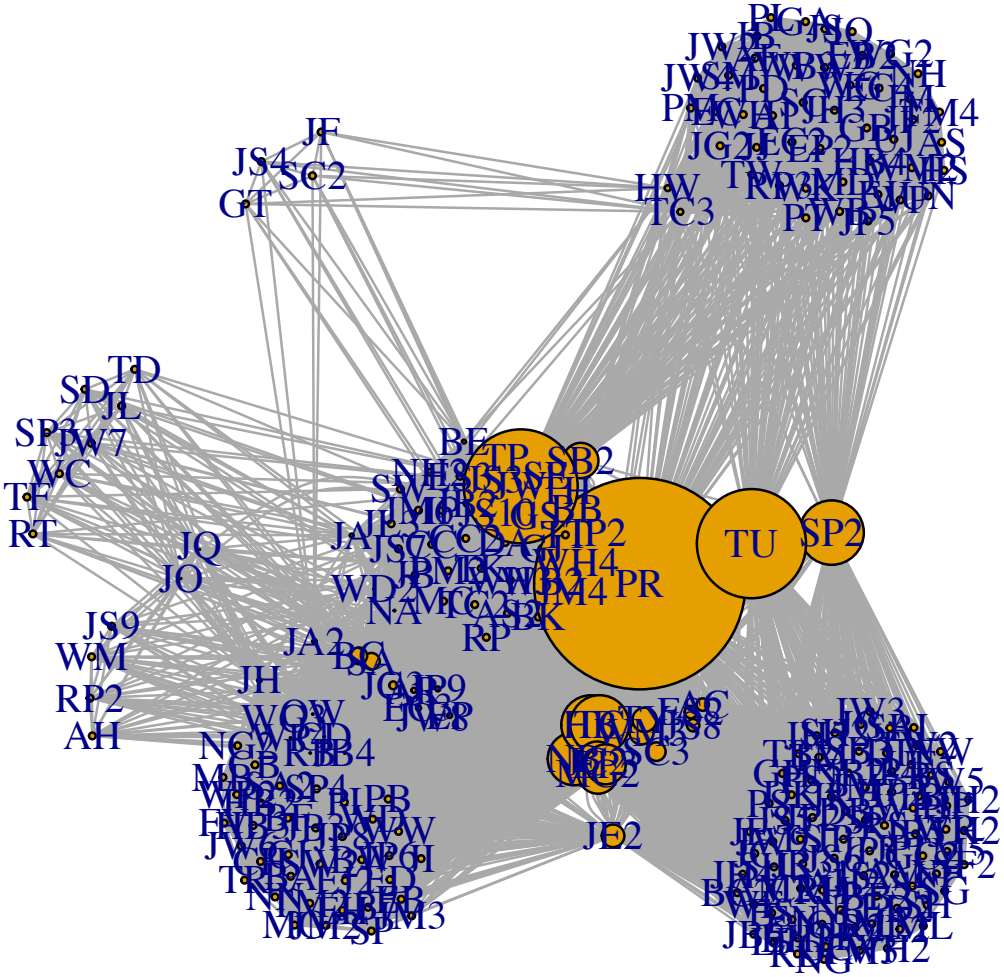
Note: in many networks there are several *equally short* paths from one node to another

betweenness: *sum of all proportions* of shortest paths from one node to another that some node is on

In this sum, each *pair of nodes* contributes

- 1 when the node is on *all* shortest paths between them
- 0 when the node is on *no* shortest paths between them
- 0.5 when it is on half of the paths, etc.

Betweenness visualised



Suspiciously between

Look for nodes with high betweenness

```
between.suspects <- sort(betweenness(apn.graph), decreasing=TRUE)
between.suspects[1:10]
```

```
       PR         JW         TU       SP2         HB         TC
4489.9016 2419.3373 2317.9788 1367.8768 1250.6579 1250.6579
       NB        JC2        MG2        EP3
1125.8952  849.0686  849.0686  849.0686
```

Centrality measures

Note:

- degree, closeness, and betweenness are *different concepts*
- there is no reason why the same nodes should maximize them all

AQ Leaders – Courier1 – Kuwaiti – Courier3 – Abbottabad household

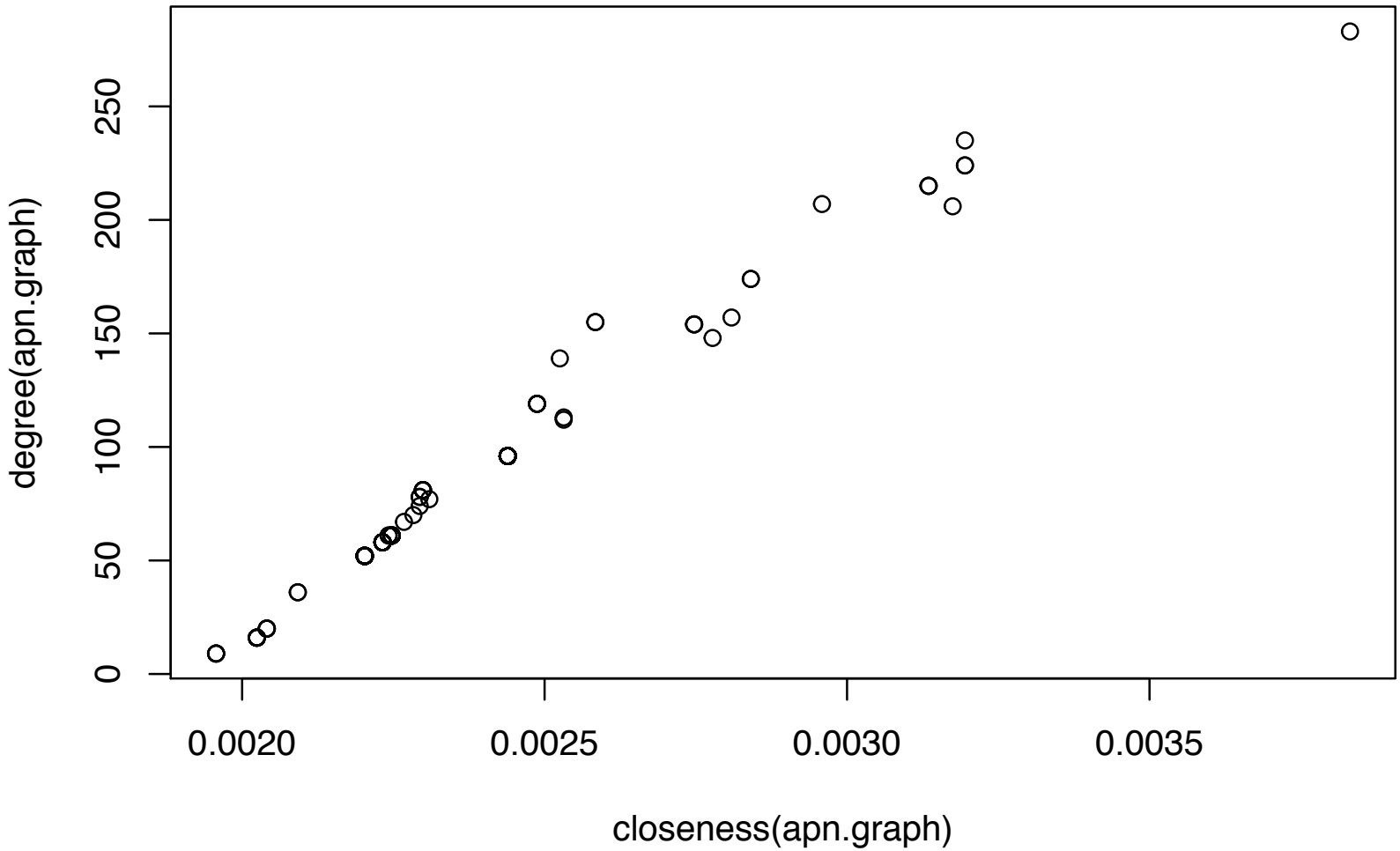
Centrality measures

Note:

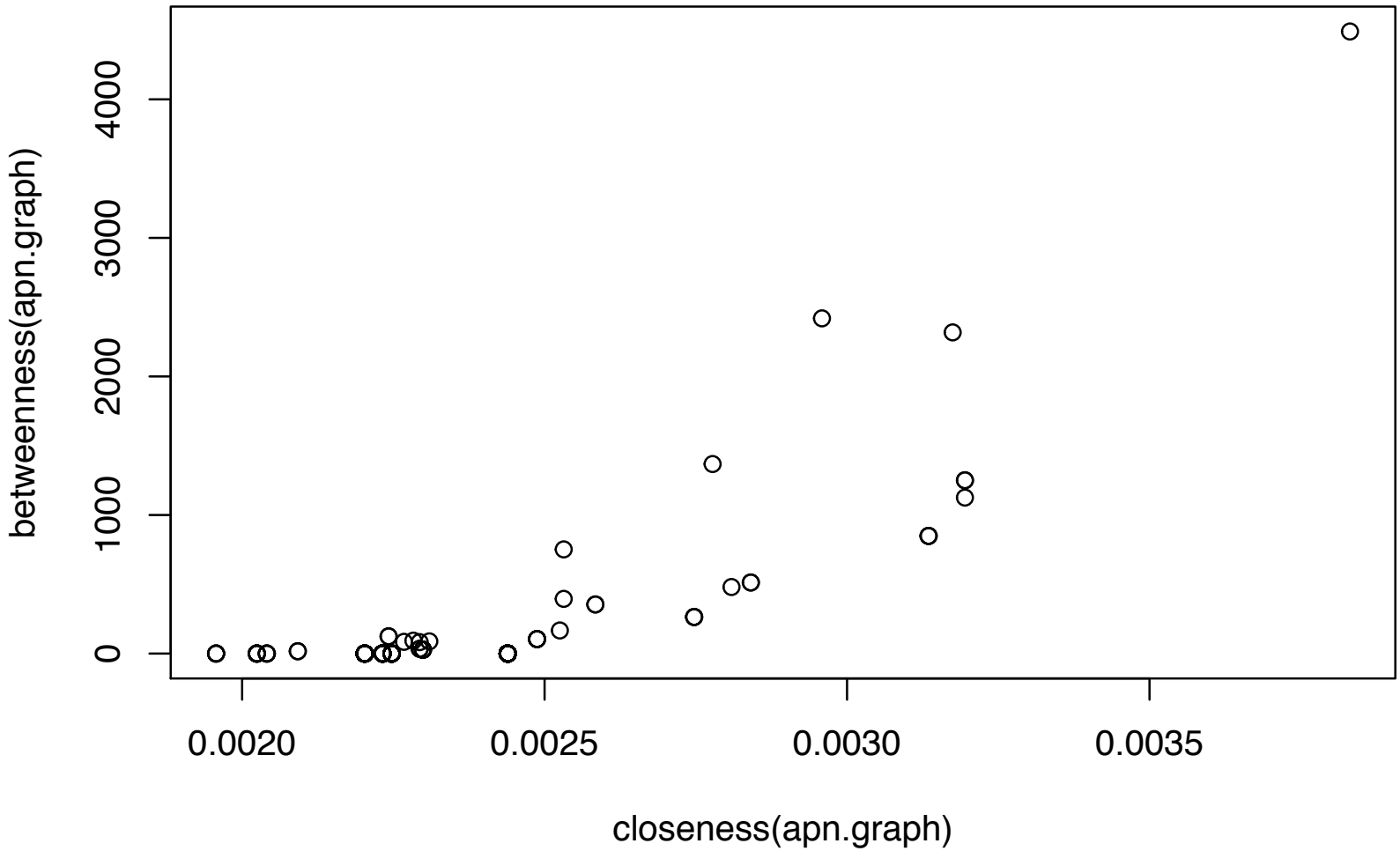
- degree, closeness, and betweenness are *different concepts*
- there is no reason why the same nodes should maximize them all

AQ Leaders – Courier1 – Kuwaiti – Courier3 – Abbottabad household
However, we do seem to have some converging evidence about the central actors in this insurgency network

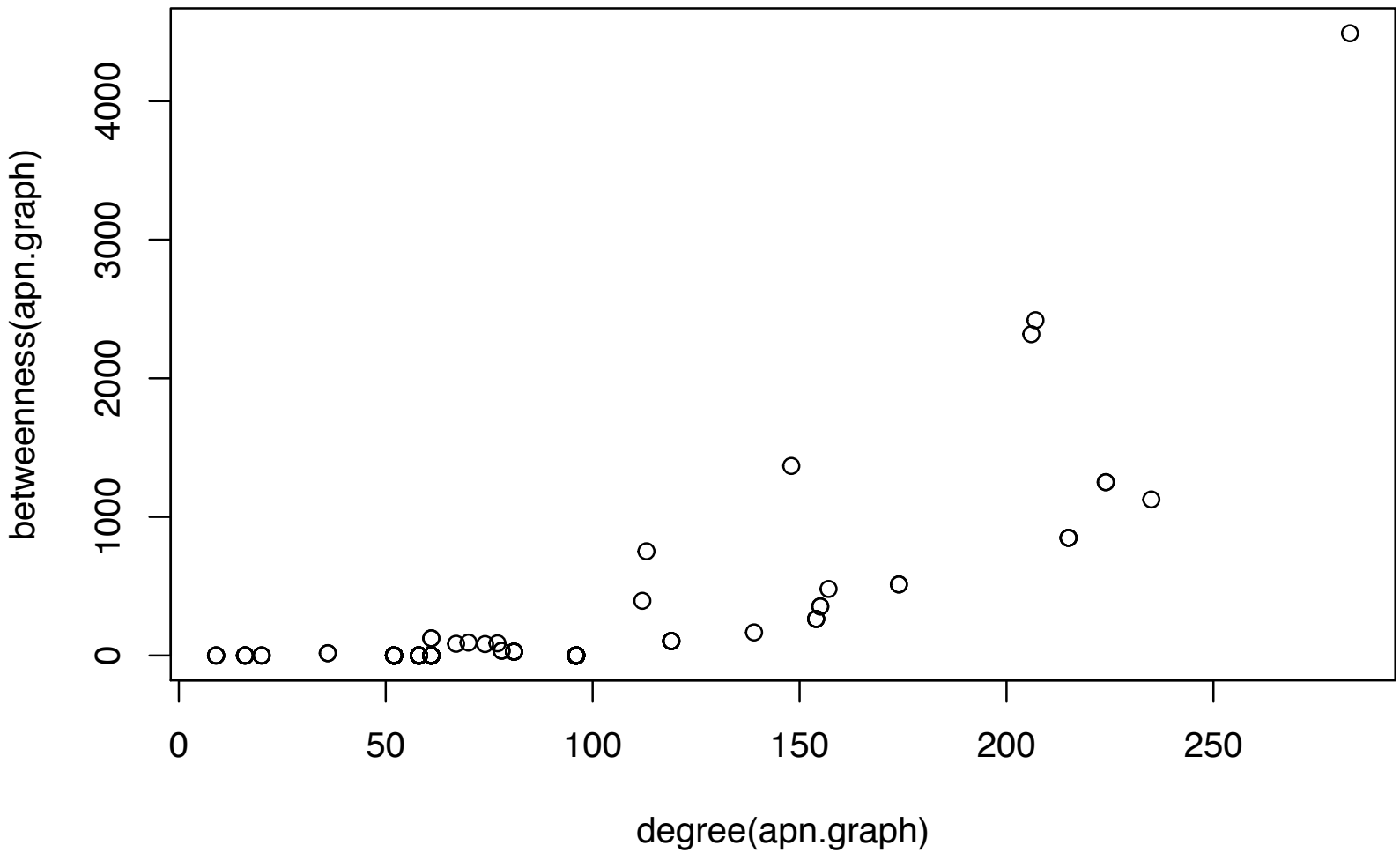
Centrality measures compared



Centrality measures compared



Centrality measures compared



Suspects

	degree	closeness	betweenness
1	PR	PR	PR
2	NB	NB	JW
3	HB	HB	TU
4	TC	TC	SP2
5	JC2	TU	HB
6	MG2	JC2	TC
7	EP3	MG2	NB
8	JW	EP3	JC2
9	TU	JW	MG2
10	WM3	WM3	EP3

PageRank as a centrality measure

We can define degree, closeness, and betweenness for directed and undirected graphs.
PageRank is a centrality (really an *importance* measure) designed only for directed graphs
Intuition:

- The more important people link to me, the more important I am

PageRank as a centrality measure

$$\text{PageRank}_j(t) = \frac{1-d}{n} + d \times \sum_{i=1}^{n} \frac{A_{ij} \times \text{PageRank}_i(t-1)}{\text{outdegree}_i}$$

- A_{ij} Whether node i follows node j
- outdegree_i makes sure i spreads one 'vote'-s worth of PageRank 'vote' around
- $\text{PageRank}_i(t-1)$ weighted by i's own PageRank

PageRank for Senators