Six degrees of separation : Tools for social network analysis in the newsroom

Sarah Cohen
The Washington Post
cohensh@washpost.com, 202-334-6259
June 2004

Sociologists, criminologists, business consultants and others have been using the decades-old field of social network analysis for years. As usual, journalists are behind.

Now some reporters are beginning to use the social network analysis tools in new ways to display and analyze connections among documents, campaign contributors, gang members and terrorists. We can also use some of the math developed for networks to identify powerful people in communities, links between seemingly unrelated companies, or crucial information brokers to interview for stories.

This handout and set of exercises will go through two tools that I have been playing with on and off for a year. It also tries to bring back to journalism some of the things I learned in taking a course on social networking theory last fall. (It was about two-third sociology theory, which we have no interest in, but one-third methods, which we do care about.)

A few basics

First, don't confuse social network analysis with the social networks that are becoming common in business and other circles. The tools we are interested in help us analyze public records and other data we collect. The popular social networks on the Web are designed to invite members in and expand a sales network or client network. They are effectively monitored listservs that simply build societies for more traditional "networking".

The analytical side of social networks is based around a powerful idea. That is, that connections among people, organizations or events are just as interesting – if not more interesting – than the "attributes" we usually see, such as sex, county, flight number or other categories.

This means thinking of your data differently. Instead of finding variables to analyze, you will look for variables that connect two people, companies or documents. A few obvious ones include members of boards of directors, cell phone records showing conversations between people, calendar entries showing meetings, crime records showing membership in gangs, or contributions to political groups.

Just remember you need a connection of some type, even if it is indirect.

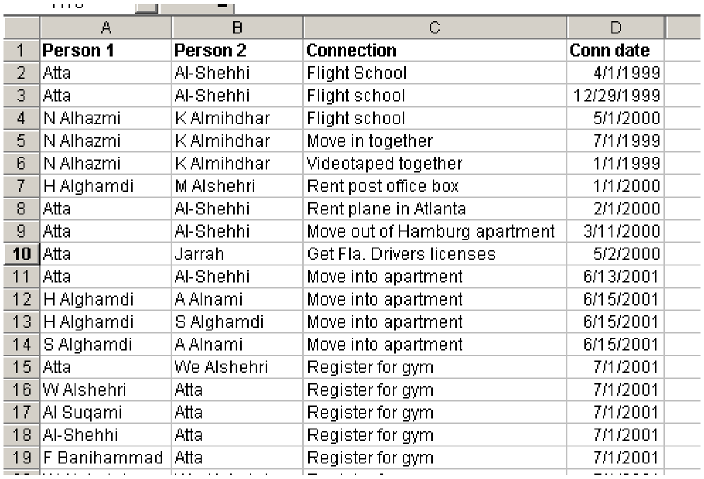
Another aspect of social network analysis is that it is based on two ways of looking at connections: in grids (or matrixes) and in graphs. Both have long and detailed theoretical histories. But the math and notation of working with data like this is difficult and makes reading textbooks nearly impossible, at least for me. I'm going to try to simplify this as much as I can, but I warn you: I understand this field only at the most superficial level, meaning that I will have difficulty making it as simple as it should be.

The final thing to remember is that the tools available to analyze social networks are new and not very commercial. I'm going to focus mainly on the tool that most academics use, called UCINet ($250). It is easier than the (free) Pajek program but has most of the tools you will need to do basic analysis. Other tools are designed solely for charting (such as Analyst 's Notebook, which I'll show you, but is too expensive and limited for widespread analytic use.)

Disk-clutter warning! UCINet creates dozens of files for each project, often with default file names. Watch what UCINet will call "output datasets". I suggest creating a separate folder for each analysis. Also, if you happen to get a chart you really like, save it – it cannot always be reproduced because UCINet usually starts at a random position on the page and the builds out from there.

Terrorist connections

The first example is documenting the connections among the terrorists who were on the four attack planes on Sept. 11, 2001. This is an abbreviated version of an analysis that was published by Valdis Krebs in the journal *Connections* after September 11 (freely available on the Web). Instead of the in-depth analysis, though, I'll use this to demonstrate how to look at social network data and some basic items we can find out about the groups.

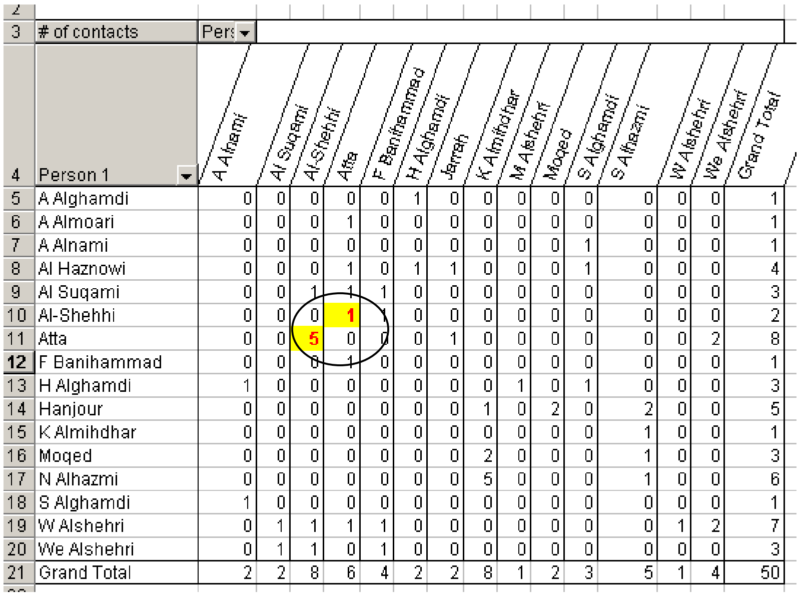


	A	B	C	D
1	**Person 1**	**Person 2**	**Connection**	**Conn date**
2	Atta	Al-Shehhi	Flight School	4/1/1999
3	Atta	Al-Shehhi	Flight school	12/29/1999
4	N Alhazmi	K Almihdhar	Flight school	5/1/2000
5	N Alhazmi	K Almihdhar	Move in together	7/1/1999
6	N Alhazmi	K Almihdhar	Videotaped together	1/1/1999
7	H Alghamdi	M Alshehri	Rent post office box	1/1/2000
8	Atta	Al-Shehhi	Rent plane in Atlanta	2/1/2000
9	Atta	Al-Shehhi	Move out of Hamburg apartment	3/11/2000
10	Atta	Jarrah	Get Fla. Drivers licenses	5/2/2000
11	Atta	Al-Shehhi	Move into apartment	6/13/2001
12	H Alghamdi	A Alnami	Move into apartment	6/15/2001
13	H Alghamdi	S Alghamdi	Move into apartment	6/15/2001
14	S Alghamdi	A Alnami	Move into apartment	6/15/2001
15	Atta	We Alshehri	Register for gym	7/1/2001
16	W Alshehri	Atta	Register for gym	7/1/2001
17	Al Suqami	Atta	Register for gym	7/1/2001
18	Al-Shehhi	Atta	Register for gym	7/1/2001
19	F Banihammad	Atta	Register for gym	7/1/2001

Here is a simple view of what the data looks like in a spreadsheet. It has two important columns – Person 1 and Person 2. As you came across items in clip searches or interviews that link any people together, you would log the connection.

It doesn't matter which order they are in, but you have to note all pairs at least once. That is, where Alghamdi, Al-Shehhi, Alnami, and others moved into an apartment together, you have to list each combination.

Now let's view this as a "grid", or a "matrix", using a Pivot Table in Excel:



3	# of contacts	Pers													
4	Person 1	A Alnami	Al Suqami	Al-Shehhi	Atta	F Banihammad	H Alghamdi	Jarrah	K Almihdhar	M Alshehri	Moqed	S Alghamdi	S Alhazmi	W Alshehri	We Alshehri	Grand Total
5	A Alghamdi	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1
6	A Almoari	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1
7	A Alnami	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
8	Al Haznowi	0	0	0	1	0	1	1	0	0	0	1	0	0	0	4
9	Al Suqami	0	0	1	1	1	0	0	0	0	0	0	0	0	0	3
10	Al-Shehhi	0	0	0	1	1	0	0	0	0	0	0	0	0	0	2
11	Atta	0	0	5	0	0	0	1	0	0	0	0	0	0	2	8
12	F Banihammad	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1
13	H Alghamdi	1	0	0	0	0	0	0	0	1	0	1	0	0	0	3
14	Hanjour	0	0	0	0	0	0	0	1	0	2	0	2	0	0	5
15	K Almihdhar	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1
16	Moqed	0	0	0	0	0	0	0	2	0	0	0	1	0	0	3
17	N Alhazmi	0	0	0	0	0	0	0	5	0	0	0	1	0	0	6
18	S Alghamdi	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
19	W Alshehri	0	1	1	1	1	0	0	0	0	0	0	0	1	2	7
20	We Alshehri	0	1	1	0	1	0	0	0	0	0	0	0	0	0	3
21	Grand Total	2	2	8	6	4	2	2	8	1	2	3	5	1	4	50

This pivot table simply counts the number of times each person has a contact with another person. It is close to the "matrix" or grid that UCINet needs.

But there are two problems here,

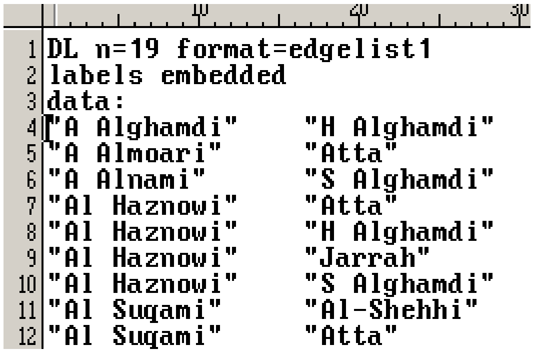
which we will fix in UCINet.

First, the number of times that Atta met with or contacted Al-Shehhi (5) is different from the number of times that Al-Shehhi contacted Atta (1). They should both be 6, since we don't care who contacted whom.

The second is that the list of people going across the page is different from the list of people going down the page. This means that the table isn't a "square matrix", and the network can't be analyzed very effectively until it is.

Although UCINet has a spreadsheet editor, which looks much like Excel, it cannot handle datasets of relationships among a set of people that are not square. That is, each actor has to be listed across the top and down the side – no one can be missing.

But it can handle lists of connections much like the one that we had originally, filling in the list with zeros wherever there is no relationship. You have to put quotes around any phrases and you have to know how many total rows and columns there will be in the final dataset. Here is what the beginning of a UCI input file looks like for our list of terrorists:



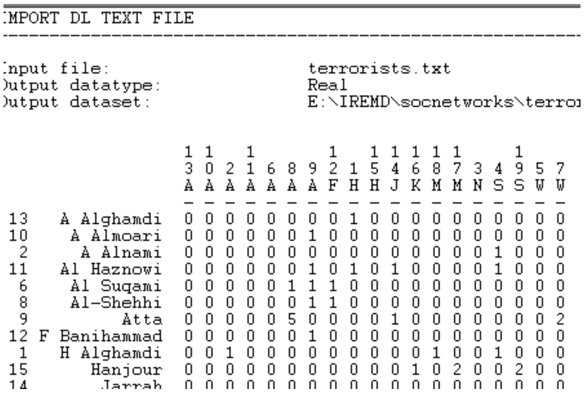
```
DL n=19 format=edgelist1
labels embedded
data:
"A Alghamdi"    "H Alghamdi"
"A Almoari"     "Atta"
"A Alnami"      "S Alghamdi"
"Al Haznowi"    "Atta"
"Al Haznowi"    "H Alghamdi"
"Al Haznowi"    "Jarrah"
"Al Haznowi"    "S Alghamdi"
"Al Suqami"     "Al-Shehhi"
"Al Suqami"     "Atta"
```

The top line says that this is an input file in a text format listing each "edge" of a connection. You have to know how many unique names you have – 19 here.

To import this list into UCINet, you would choose **Data**, **Import**, **DL** and then provide the file name of your list.

After importing, you get a grid that looks very similar to what you saw in the spreadsheet. But there are now 19 rows and 19 columns, each with the same names. This is called a "square matrix" or an "adjacency matrix" in the menus.

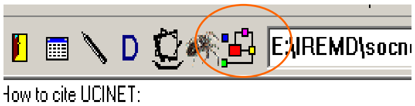
It is now saved as a UCINet file called "terrorists".



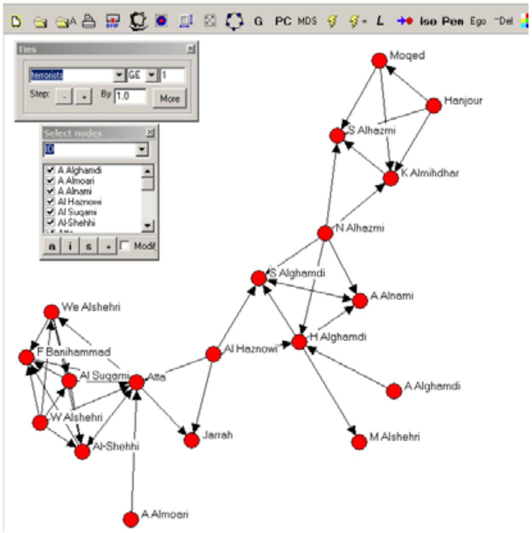
```
IMPORT DL TEXT FILE
-----------------------------------------------------------------

Input file:                    terrorists.txt
Output datatype:               Real
Output dataset:                E:\IREMD\socnetworks\terro

                           1 1   1         1   1 1 1 1 1       1
                           3 0 2 1 6 8 9 2 1 5 4 6 8 7 3 4 9 5 7
                           A A A A A A A F H H J K M M N S S W W
                           - - - - - - - - - - - - - - - - - - -
 13   A Alghamdi           0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0
 10    A Almoari           0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0
  2     A Alnami           0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0
 11   Al Haznowi           0 0 0 0 0 0 1 0 1 0 1 0 0 0 0 1 0 0 0
  6    Al Suqami           0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0
  8    Al-Shehhi           0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0
  9         Atta           0 0 0 0 0 5 0 0 0 0 1 0 0 0 0 0 0 0 2
 12 F Banihammad           0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0
  1   H Alghamdi           0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0
 15      Hanjour           0 0 0 0 0 0 0 0 0 0 0 1 0 2 0 0 2 0 0
 14       Jarrah           0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

Drawing the charts

In UCINet, the drawing program is called NetDraw. You get to it by pressing the drawing button, then opening the file you just imported:



Using the drawing program, here is how the people are connected:

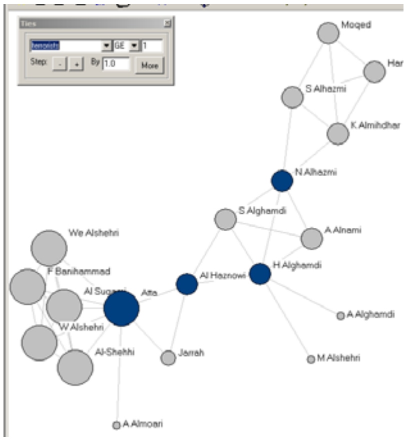


The arrows show you which direction the connection went and are irrelevant to us. (They won't always be irrelevant. For example, if you had cell phone or payment records, you might care that one person called or paid the other, not the other way around.)

The arrows can be turned off using the icon with the little arrow on it. The position of the graphic is set with the other buttons on the top. (If you wanted to use the number of contacts then you could try some other options. But usually we'll stick with the lightning bolt with the equal sign.) The length of the lines is meaningless. The program places the dots where they have to be on the page to minimize the number of crossed lines. (If you want to watch it try, use the menu item Layout, Spring embedding, then choose Distances + Node Repulsion and a starting position of Gower Scaling. Then up the number of iterations to something much higher, like 1000.)

Now we're at a frustrating point with UCINet. What I would really like is to see which terrorist was on which airplane. Although it has options for such attribute data, I have never successfully imported this kind of data into the programs.

We can do a little analysis right here to see a few things that are obvious in this graph but may be less clear in others. We can find what's known as cutpoints –key players in the network who, if removed, would mean that all connections between groups would be gone. This is a picture of the cutpoints in the terrorist network:

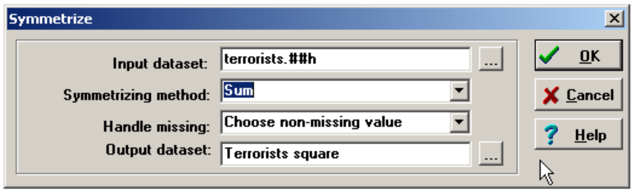


While I naturally see some of the cutpoints in a graphic, I've found that I usually don't notice them all. Showing them in this way can help.

More analysis in UCI Net

The next step is to describe the network with some standard measures much like a simple crosstab or descriptive statistics analysis on other kinds of data.

To do that, you'll go back to UCINet, and then make some changes to the dataset to make it easier to analyze.



Symmetrize

Input dataset: terrorists.##h

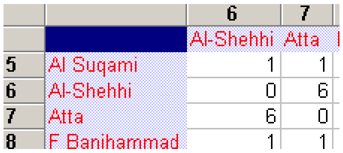
Symmetrizing method: Sum

Handle missing: Choose non-missing value

Output dataset: Terrorists square

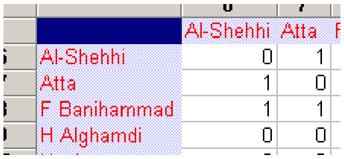
Going back to UCINet, you will have to do two things to make this dataset work for many of the standard analysis methods: You have to make it symmetrical, meaning that you can turn it upside down or sideways and still get the same answers; and you will normally make it dichotomous, meaning that the number of contacts between terrorists is ignored and only whether or not there is a contact is saved. This is done through procedures under the **Transform** menu item.

After making it symmetrical, the differences that we saw in the spreadsheet disappear:



		6	7
		Al-Shehhi	Atta
5	Al Suqami	1	1
6	Al-Shehhi	0	6
7	Atta	6	0
8	F Banihammad	1	1

And after making it dichotomous, all of the numbers are 1 or 0:

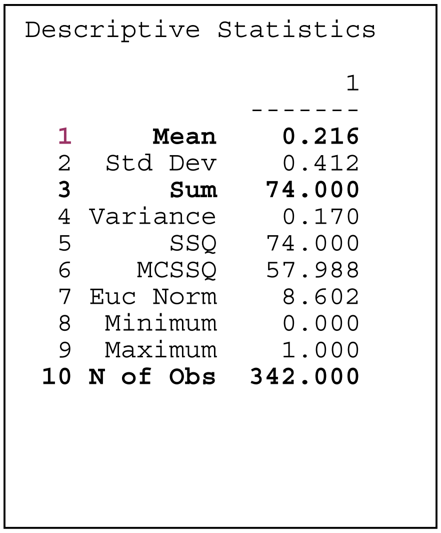


	Al-Shehhi	Atta
Al-Shehhi	0	1
Atta	1	0
F Banihammad	1	1
H Alghamdi	0	0

Now many of the standard methods of analyzing networks will be easier to interpret. (Note: There is a specific problem in analyzing “unconnected” networks – those that have groups or individual items not joined by a line to the big graph. You usually have to eliminate them from your dataset before you can go further. This network is fully connected so no further changes are necessary.)

Basic statistics about a network:

The first thing, like in statistical analysis, is to get a general idea of your network and how it is structured. Through UCINet’s **Univariate** procedure, you can get the “density” – the number of lines present compared to the number of lines possible in the graph.



```
Descriptive Statistics

                        1
                  -------

 1       Mean       0.216
 2    Std Dev       0.412
 3        Sum      74.000
 4   Variance       0.170
 5        SSQ      74.000
 6      MCSSQ      57.988
 7   Euc Norm       8.602
 8    Minimum       0.000
 9    Maximum       1.000
10   N of Obs     342.000
```

Because all of the numbers in our graph are 0 and 1, the mean tells you what portion of the total lines are present: 21.6 percent. (You would get the same answer if you used the menu item **Network**, **Network Properties**, **Density**). For a small network, this is a pretty low number. One problem in network analysis is that it is not “scale-free”. That is, it’s a lot easier to know everyone in a room than everyone in a concert hall. So density tends to go down as the size of the network grows.

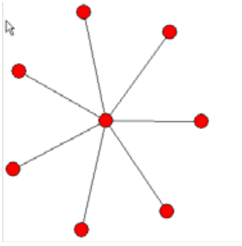
The other bolded numbers tell us that there are 74 lines connecting people, out of a possible 342.

We can also look at how closely linked each person is to everyone else, looking at “degree centrality” under the Network, Centrality, Degree menu. (I erased some people to save space):

```
                                     1            2            3
                                Degree    NrmDegree        Share
                          ------------ ------------ ------------
  7             Atta             8.000       44.444        0.108
  9       H Alghamdi             6.000       33.333        0.081
  6        Al-Shehhi             5.000       27.778        0.068
  8  F Banihammad                5.000       27.778        0.068
 19    We Alshehri               5.000       27.778        0.068
 11           Jarrah             2.000       11.111        0.027
  2        A Almoari             1.000        5.556        0.014
 13      M Alshehri              1.000        5.556        0.014
```

Mohammed Atta is the most closely linked person in the network. It means that of all the possible connections he can have, he has almost half of them – 44 percent. At the bottom of the page, it also shows you totals for the network as a whole. (For a symmetrical network with only ones and zeros, this is the same thing as the density information we just got).

```
DESCRIPTIVE STATISTICS

                                 1            2            3
                            Degree    NrmDegree        Share
                      ------------ ------------ ------------
  1       Mean               3.895       21.637        0.000
  2    Std Dev               1.774        9.855        0.000
  3        Sum              74.000      411.111        0.000
  8    Minimum               1.000        5.556        0.000
  9    Maximum               8.000       44.444        0.000

Network Centralization = 25.49%
```



The “Network centralization” measure here and in most other “centrality” routines in UCINet compares this graph with one that is completely centralized – one that looks like a dot with a bunch of spokes coming out of it (Comparable figures are: The mean “degree centrality” in this example is 25, and the “Network centralization” is 100).

Again, it is a relatively low number, at 25 percent.

This is great for checking how many *direct* links there are in a network. But many network properties are really measuring how important various actors are, and how far you have to go to get to anyone in the network. There are two other main measures: Closeness and Betweenness.

Closeness:

One way to look at how close everyone is in the network is to check the average number of steps you have to go to get to everyone else. This takes into account all the ties in the network, not just those right next to each actor.

Here is an example of the “closeness” measures for some of the actors in the terrorist network:

```
                               Farness   nCloseness
```

```
                       ------------ ------------
  9   H Alghamdi         38.000       47.368
  4   Al Haznowi         38.000       47.368
 16   S Alghamdi         40.000       45.000
  7         Atta         42.000       42.857
 15    N Alhazmi         45.000       40.000
 11       Jarrah         48.000       37.500
```

“Farness” is the sum of the number of steps it takes to get to each other person. Its reverse is closeness, and the “nCloseness” is how close the actor is from the smallest possible number, or if everyone were one step away. (In this graph, the lowest number is 18 – one less than the number of actors. 18/38 = 47.368.) This means that even when an actor can’t reach others in one step, he may be close in just two steps. So while Mohammed Atta is connected to the most number of people directly, he is pretty far from the people to whom he had no connections. His closeness score is lower than the people smack in the middle of our diagram.

Betweenness:

The last standard measure of centrality that I vaguely understand is “Betweenness”. This takes into account the importance of an actor in a network. It’s particularly important to look at betweenness if you are trying to find brokers – people like secretaries, operatives or others whom you might not notice but act as gatekeepers to important people in the network. They may be excellent people to interview since they have the most opportunity to hear, directly and indirectly, from others in the group. They can be great sources. They also may be roadblocks. Again, I’ve deleted some rows to compact the printout.

```
Un-normalized centralization: 1190.000

                                  1            2
                        Betweenness nBetweenness
                       ------------ ------------
  4   Al Haznowi         80.000       52.288
  7         Atta         77.000       50.327
  9   H Alghamdi         60.000       39.216
  8 F Banihammad          0.000        0.000
 18   W Alshehri          0.000        0.000
 19  We Alshehri          0.000        0.000


                Betweenness nBetweenness
               ------------ ------------
  1     Mean       17.368       11.352
  2  Std Dev       27.637       18.063
  3      Sum      330.000      215.686
  8  Minimum        0.000        0.000
  9  Maximum       80.000       52.288

Network Centralization Index = 43.21%
```

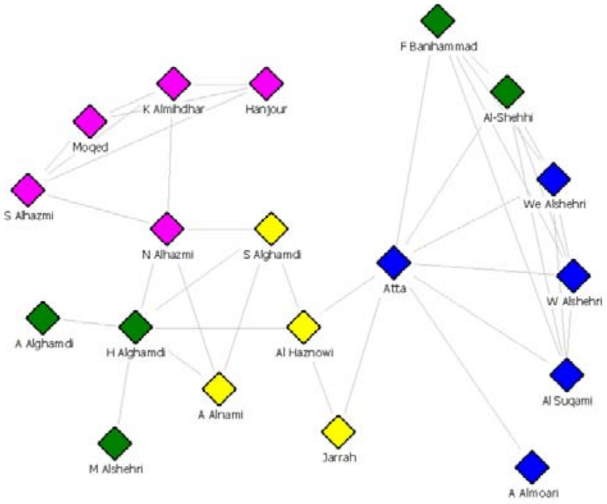
This says that Mohammed Atta lies between two halves of the network – to get to half of the people, the others have to go through him. It’s a measure of each person’s power in the network.

This is a relatively high level of “betweenness” for a small network. It’s often the reverse of the other centralization measures, particularly centrality.

There are other measures of a network, but these are the basic ones that are well understood and generally can be used in reporting to find interesting subjects for stories or characterizations of a group of people.

Terrorist networks in Analyst's Notebook

Another tool we have used to make charts of networks is "Analyst Notebook". This is a completely different type of tool – it makes charts and does very little else, but is very flexible in using many different types of information, including other data elements you might have on the person. Here is an example in Analyst Notebook of the terrorist network:



With Analyst Notebook, you only have two options for displaying the data automatically. This example works well in it, but most don't. (The reason is that it is so clearly partitioned among the terrorists.)

You can't do any other analysis but there are some things you can do here that you can't do as easily in UCINet. The most obvious is that it is easy create colors for another variable you import. In this case, it is by which plane they were on during the Sept. 11 attacks.

It is also to find paths between actors and to add other information to your chart.

But the analysis is very limited and the automatic charts on a more realistic network often take a great deal of hand-editing.

(Analyst Network is very expensive and has limited usefulness outside its charting capabilities, which are mainly hand-drawn. But it is good at combining timelines with networks and does let you look at data in several different ways.)

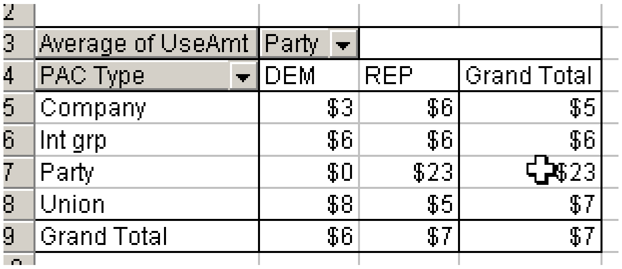
Another example: Campaign contributions

So far, we've been dealing with a classic "adjacency" network – one in which we know contacts among everyone and are looking for patterns in who knows whom. But more frequently, we deal with what's called "affiliation networks" – a branch of social network analysis that is less understood and has fewer methods of analysis.

Most public records we'll encounter are actually rectangular rather than square data – it lists every company, say, across the top and each person associated with that company down the side. The Post used this method when we received information about officials who had signed visas for language schools in California.

It is always possible to remove the intermediary and draw the links directly to the people in a database like Access. But in social network analysis, you usually let the program do it for you.

Here is an example of data downloaded from the FEC on campaign contributions to members of the House Energy Committee's subcommittee on environment and hazardous waste. There are 12 Democrats and 16 Republicans on the committee. I chose the 20 largest PAC givers but ended up combining two different Republican Party PACs into one, leaving 19.



2				
3	Average of UseAmt	Party ▾		
4	PAC Type ▾	DEM	REP	Grand Total
5	Company	$3	$6	$5
6	Int grp	$6	$6	$6
7	Party	$0	$23	$23
8	Union	$8	$5	$7
9	Grand Total	$6	$7	$7

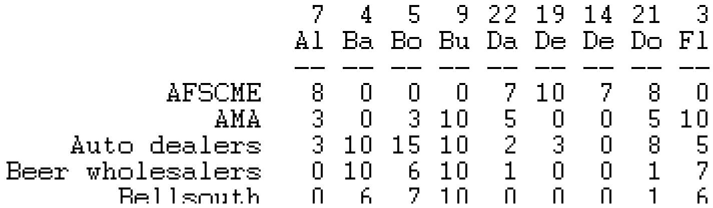
It's obvious from these averages that the only party PAC money went to Republicans. But on average, Labor PACs gave $5,000 to Republicans and $8,000 to Democrats. And interest group PACs – which usually call themselves nonpartisan – gave equally to both parties.

Think of the data as "affiliation" or "2-mode" data – it can be viewed as a rectangular grid with PACs down the side and candidates across the top (or the other way around). That is, if you made a pivot table of candidate by PAC, the columns would not refer to the same thing as the rows.

```
DL nr=19 nc=28 format=edgelist2
labels embedded
data:
"NRCC / RNC"      "Wilson"      81
"NRCC / RNC"      "Shimkus"     75
"NRCC / RNC"      "Fletcher"    72
"NRCC / RNC"      "Bass"        70
"Auto dealers"    "Bono"        15
"NRCC / RNC"      "Wilson"      15
"NRA"             "Fletcher"    14
"NRCC / RNC"      "Shimkus"     14
"Laborors Intl"   "Stupak"      13
```

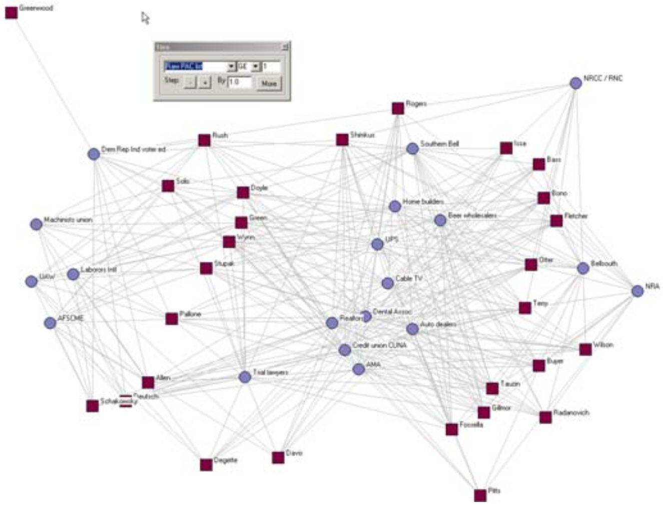
You import the data the same way into UCINet, except you change the top row a little to indicate that you have two elements, rows and columns. The other change is that you add a number next to each entry to show how much is given.

Here is what the import looks like when it is done (you can control the labels to show more of the name across the top at the expense of being able to see all of them)



```
                     7  4  5  9 22 19 14 21  3
                    Al Ba Bo Bu Da De De Do Fl
                    -- -- -- -- -- -- -- -- --
            AFSCME   8  0  0  0  7 10  7  8  0
               AMA   3  0  3 10  5  0  0  5 10
      Auto dealers   3 10 15 10  2  3  0  8  5
  Beer wholesalers   0 10  6 10  1  0  0  1  7
         Bellsouth   0  6  7 10  0  0  0  1  6
```

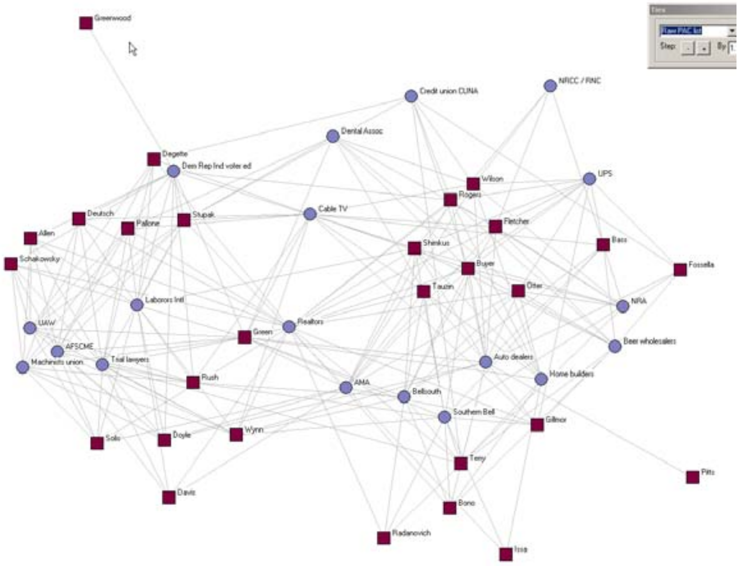
The numbers in the middle now refer the amount given (in thousands) to each candidate by each PAC during the 2002 election.



The original graph (or, to use the technical word, the hairball) that UCINet automatically builds doesn't tell us much.

Nearly every PAC gives to every candidate on the list so virtually every possible line is drawn. But we haven't yet tried to distinguish between a lot of money and a little.

If you notice the little box on top, it allows for cutoffs that will determine when the line is used to connect the dots. There are a couple of ways to choose a cutoff – an average ($7,000 from our table above), a standard, or a standard deviation. I chose a standard – at the time, the limit for each election was $5,000. So if a PAC gave the maximum for either a primary or general election, or totaled that amount, I'll count it as strong support:

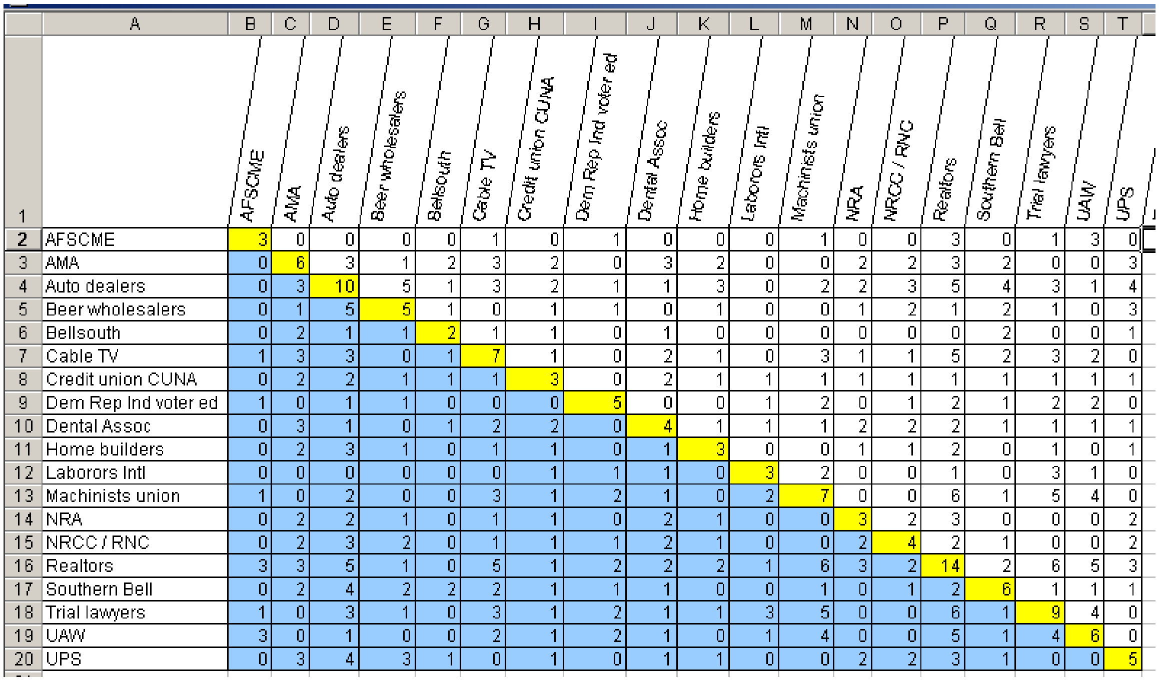


Now it's a little easier to read, and it's a little easier to see the patterns: Unions and trial lawyers are clearly placed on one side with Democrats; the NRA, United Parcel Service, Beer Wholesalers and the party committees are clearly placed on the other side with the Republicans.

If you were to enter the affiliation data of each candidate's party, you would find that Greenwood is the only oddity – a Republican congressman on the Democratic side of the chart. It turns out he has said he does not accept PAC money, but must have accepted some from the Teamster-sponsored voter drive fund. (Joe Pitts, the other lawmaker sticking out, was virtually uncontested in this election.)

It would be nice to see what other kinds of groups might be natural splits in the network. We can ask the graphics program to do that but it gets very confused when there is a mix of companies and people.

One common way to reduce the graph is to first turn the values in the middle of the grid into yes-or-no questions (dichotomize again), then turn the grid on its side and multiply it by itself. I chose a cutoff under **Transform**, **Dichotomize**, of the largest amount most PACs can give in a 2-year period: $10,000. Then I used the menu item **Data**, **Affiliations**, **Crossproduct** to turn it on its head and multiply, saying that I wanted to keep the rows (PACs) not the columns (Candidates).



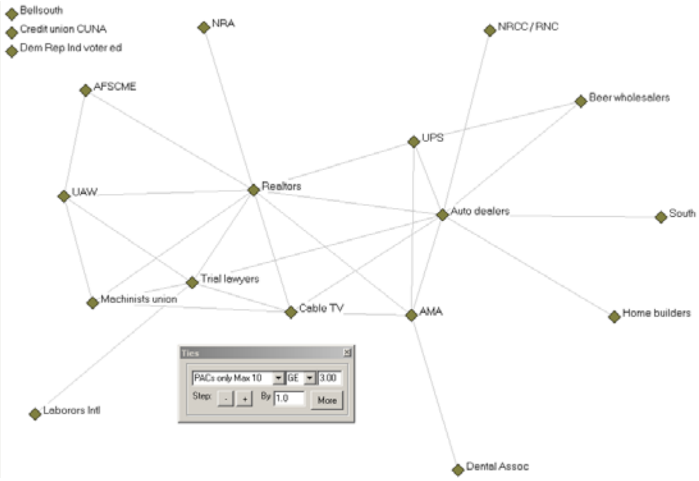
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1		AFSCME	AMA	Auto dealers	Beer wholesalers	Bellsouth	Cable TV	Credit union CUNA	Dem Rep Ind voter ed	Dental Assoc	Home builders	Laborors Intl	Machinists union	NRA	NRCC / RNC	Realtors	Southern Bell	Trial lawyers	UAW	UPS
2	AFSCME	3	0	0	0	0	1	0	1	0	0	0	1	0	0	3	0	1	3	0
3	AMA	0	6	3	1	2	3	2	0	3	2	0	0	2	2	3	2	0	0	3
4	Auto dealers	0	3	10	5	1	3	2	1	1	3	0	2	2	3	5	4	3	1	4
5	Beer wholesalers	0	1	5	5	1	0	1	1	0	1	0	0	1	2	1	2	1	0	3
6	Bellsouth	0	2	1	1	2	1	1	0	1	0	0	0	0	0	0	2	0	0	1
7	Cable TV	1	3	3	0	1	7	1	0	2	1	0	3	1	1	5	2	3	2	0
8	Credit union CUNA	0	2	2	1	1	1	3	0	2	1	1	1	1	1	1	1	1	1	1
9	Dem Rep Ind voter ed	1	0	1	1	0	0	0	5	0	0	1	2	0	1	2	1	2	2	0
10	Dental Assoc	0	3	1	0	1	2	2	0	4	1	1	1	2	2	2	1	1	1	1
11	Home builders	0	2	3	1	0	1	1	0	1	3	0	0	1	1	2	0	1	0	1
12	Laborors Intl	0	0	0	0	0	0	1	1	1	0	3	2	0	0	1	0	3	1	0
13	Machinists union	1	0	2	0	0	3	1	2	1	0	2	7	0	0	6	1	5	4	0
14	NRA	0	2	2	1	0	1	1	0	2	1	0	0	3	2	3	0	0	0	2
15	NRCC / RNC	0	2	3	2	0	1	1	1	2	1	0	0	2	4	2	1	0	0	2
16	Realtors	3	3	5	1	0	5	1	2	2	2	1	6	3	2	14	2	6	5	3
17	Southern Bell	0	2	4	2	2	2	1	1	1	0	0	1	0	1	2	6	1	1	1
18	Trial lawyers	1	0	3	1	0	3	1	2	1	1	3	5	0	0	6	1	9	4	0
19	UAW	3	0	1	0	0	2	1	2	1	0	1	4	0	0	5	1	4	6	0
20	UPS	0	3	4	3	1	0	1	0	1	1	0	0	2	2	3	1	0	0	5

This is a little hard to read, especially if you are looking at this in black and white. But everything above the yellow (lightly) shaded cells is a mirror image of everything below it. That lightly shaded area, called the "diagonal" has a specific meaning: It is the number of candidates to whom each PAC gave the maximum allowed by law. (Notice that you can't just add up the row – the same combinations are counted, by, say, the Realtors and the UAW.)

By lining up the PACs against each other you can see the number of candidates they gave to in common. For example, AFSCME, the public employees union, gave the maximum amount to none of the same candidates that the American Medical Association, but it gave the maximum to the same three candidates as the United Auto Workers' PAC.

(There is another way to compare them, using correlation coefficients, but we'll skip that this time. Just remember it is a good way to work with affiliation data that has values like dollars in it and that you can find interesting groups by using that analysis. It's under **Tools**, **Similarities**... in the menu. In fact, it works much better to help identify the groups in this example, so you might want to try it on your own.)

After making a graph of this result we can cycle through the number of candidates each PAC has in common using the same box we used before. Here is how it looks if we insist that, in order to draw a line, the PAC have at least three candidates in common with another PAC:



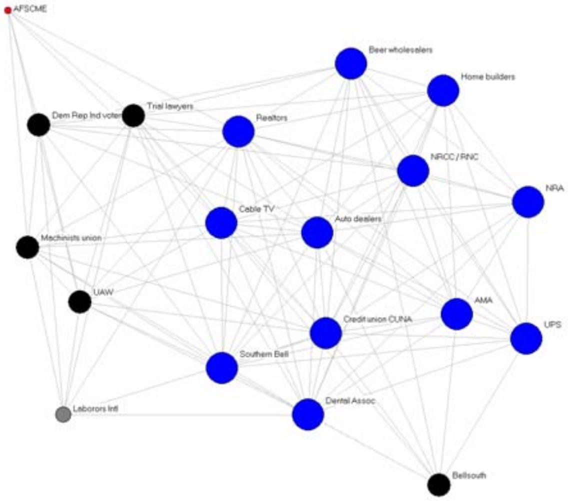
PACs only Max 10 GE 3.00
By 1.0

By saying that each PAC have at least three candidates in common, we've removed a lot of the crossed lines and can see a better pattern. We can also see that three PACs don't share three candidates with any others.

The NRA has also moved closer to the center of the map, and the business interests are more clearly clustered together. An examination of the data might show other patterns, such as incumbency.

Another way to identify groups is through a method called "k-cores". This routine tries to find groups with a certain number of ties in common. Then it relaxes the restriction as it adds more. It is useful for finding a core group inside a large network.

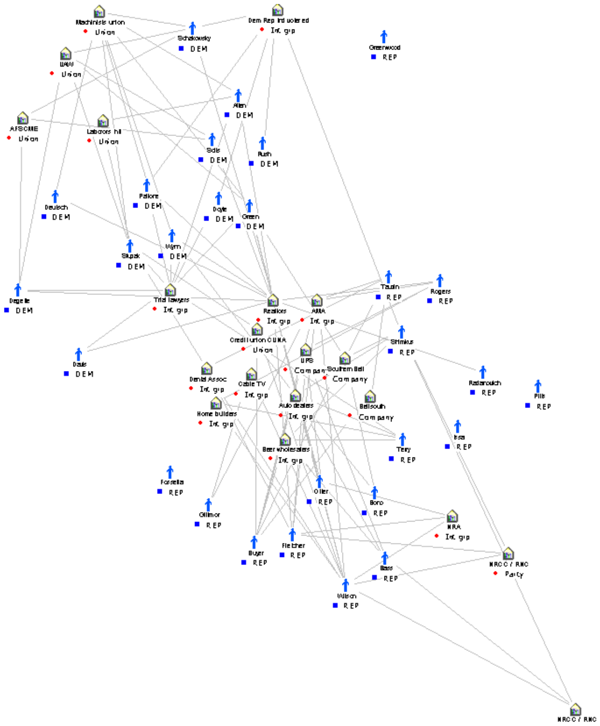
Here is a k-core analysis of the PAC list (with at least one candidate in common used as the basis for the lines.):



Although I'm not going to go through it here, the other item that I've found useful is called "factions". It blocks out the grid, trying to find groups of PACs, in this case, that are more similar within the groups than across them. They can identify interesting coalitions.

PACs and contributions in Analyst's Notebook:

Before you get too excited about how much better Analyst Notebook is for displaying information like the terrorists, consider this chart:

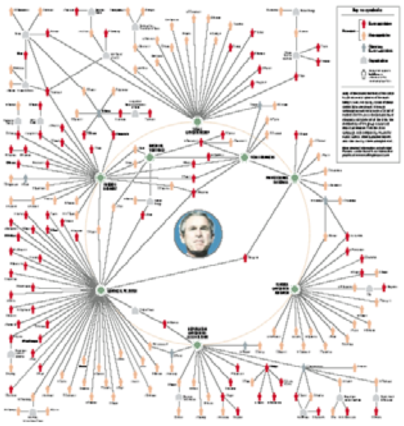


This is the best I can do in Analyst Notebook in automatically creating the placement of items in the chart, even after deleting any links that add up to less than $10,000.

Although there might be something of interest in here, it seems very difficult to extract it. Any further adjustments would have to be made by hand.

Back to the real world

In 2004, The Washington Post published a story based in part on an analysis of the original Bush Pioneers – 246 people who had raised at least $100,000 each for the president's first campaign run. We were interested in three questions: Who are they, how did they come to support the campaign at such an involved level, and what did they get in return? As a side analysis, we wondered whether the campaign had diversified its base and recruited traditional Republican fundraisers for the 2004 bid.



Here is a thumbnail of the chart (or large hairball) that eventually ran in the paper. (You can see an interactive version of the chart at http://www.washingtonpost.com/wp-srv/politics/pioneers/pioneers_spheres.html (registration required -- sorry).

At first, I had grand illusions that social network software would help in this project. All I had to do was collect the names, find out a few facts about each, and the chart would create itself.

Yeah, sure.

It turns out that thinking for social network analysis is different than thinking for databases, and it turns out that neither UCINet nor Analyst Notebook helped much in the project. Thinking about it, the reason is obvious.

To get the software to show us the connections, we would have had to have known one of two things. For the traditional, one-mode analysis, we would have had to have known how each of the 246 members of the network were connected to each of the others. We'd obviously never know that. To do a two-mode, or affiliation, analysis, we had to identify the "affiliations". Although company connections helped, it masked some of the biggest connections, such as fellow baseball team owners or members of the Republican Governors' Association.

As it turned out, the process of collecting that information – categorizing each fundraiser into one or many affiliation groups, like a Texas gubernatorial appointee, a family friend, a relative, an investor in a Bush oil venture or the like, created a database that required no social network analysis software to analyze.

I ended up drafting several charts in Analyst Notebook, but it could have been done just as easily in any drawing tool. It had to be done essentially by hand. Then it had to be traced by our artists, and they worked on making it more effective while keeping the basic structure. All of the items had to be hand-checked and the online version had to be coded by hand by a freelancer hired by the web site.

(This was a little harder than other projects, because we were dealing with what sociologists call and "ego network" – one that is driven by contacts of a single person. They have many fewer analytical techniques.)

The lesson? I think that we learned that the concepts help us envision new ways to look at projects, but the tools may not always make it any easier.