# SNA Basics #1
## One-Mode Social Network Data

This lab's purpose is to introduce readers to one-mode social network data and how we can analyze it using the social network programs UCINET (and NetDraw), Pajek, and R. For UCINET and Pajek, commands appear in the margins with an accompanying explanation in the text. The instructions for appear in a separate document (`SNA Basics #1 (R).pdf`) and are embedded in the text itself. The data used in this example can be found in the file, `SNA Basics #1 (Data).zip`. Before illustrating how to work with one-mode data, a brief discussion of this type of social network data is in order.

One-mode networks consist of a single set of actors, such as people, groups, denominations, corporations, nation-states, etc. They differ from two-mode networks, which consist of two sets of different actors, or one set of actors and one set of events or affiliations. Traditionally, one-mode social network data have been recorded using matrices where a value of "1" or greater in a common cell indicates a tie between two actors and a "0" indicates the absence of a tie. Consider, for example, the following matrix, which presents data that Samuel Sampson collected of novices at a monastery shortly after the end of Vatican II (Figure 2.15 in the book). During Sampson's period of observation, a "crisis in the cloister" occurred in response to some of the changes proposed by the Second Vatican Council (Vatican II). This led to the expulsion of four novices and the voluntary departure of several others. Based on his observations, Sampson partitioned (i.e., sorted, divided) the novices into four groups: (1) the young turks, (2) the loyal opposition, (3) the outcasts, and (4) the neutrals. The young turks arrived later and questioned practices of the monastery, which members of the loyal opposition defended. The outcasts were novices that weren't accepted into the larger group, while the neutrals were those who didn't take sides in the debate over the monastery's practices. Most of the loyal opposition had attended a seminary, "Cloisterville," prior to their arrival at the monastery.

```
                   1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18
                   Ra Bo Am Be Pe Lo Vi Wi Jo Gr Hu Bo Ma Al Am Ba El Si
                   mu na br rt te ui ct nf hn eg gh ni rk be an si ia mp
                   al ve os ho  r  s or ri  B or    fa    rt  d  l  s li
                    d nt  e ld          d os  y    ce             s ci
                      ur                  co                         us
                       e
                   -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
 1     Ramuald      0  1  1  0  3  0  0  0  0  0  0  0  0  0  2  0  0  0
 2  Bonaventure     0  0  1  0  3  2  0  0  0  0  0  0  0  0  0  0  0  0
 3     Ambrose      0  1  0  0  0  0  3  2  0  0  0  0  0  0  0  0  0  0
 4    Berthold      0  1  2  0  3  0  0  0  0  0  0  0  0  0  0  0  0  0
 5      Peter       0  3  0  1  0  2  0  0  0  0  0  0  0  0  0  0  0  0
 6      Louis       0  2  0  0  0  0  3  0  0  0  1  0  0  0  0  0  0  0
 7     Victor       0  0  1  2  3  0  0  0  0  0  0  0  0  0  0  0  0  0
 8    Winfrid       0  0  0  0  0  0  0  0  3  2  0  0  1  0  0  0  0  0
 9   John Bosco     0  0  0  0  0  0  0  1  0  0  2  0  0  0  0  3  0  0
10    Gregory       0  0  0  0  0  0  0  2  3  0  0  0  1  0  0  0  0  0
11      Hugh        0  0  0  0  0  0  0  1  3  0  0  2  0  0  0  0  0  0
12   Boniface       0  0  0  0  0  0  0  2  0  3  0  0  1  0  0  0  0  0
13      Mark        0  0  0  0  0  0  0  1  0  3  0  0  0  2  0  0  0  0
14     Albert       0  0  0  0  0  0  0  0  0  3  0  1  2  0  0  0  0  0
15     Amand        0  2  0  0  0  0  0  0  0  0  0  1  0  0  0  0  0  3
16     Basil        0  0  0  0  0  0  0  3  0  0  0  0  0  2  0  1  2
17     Elias        0  0  0  0  0  0  0  0  0  1  0  0  0  0  0  2  0  3
18   Simplicius     0  0  0  0  0  0  0  0  0  1  0  0  0  0  0  2  3  0
```

An advantage of using matrices is that they are highly intuitive. For example, it is easy to see that Ramuald shares ties of strength 1with Bonaventure and Ambrose, a tie of strength 3 with Peter, and a tie of strength 2 with Amand. Unfortunately, social network files stored as matrices tend to be quite

large. Why? Because not only do you have to record ties, you have to record the absence of ties. That is why some programs use edge lists instead of matrices. With edge lists we only need to record the ties. For example, take a look at the following edge list, which contains the same data as above. It first indicates the number of actors/vertices (18) and then lists their labels (the numbers to the right of the labels are the visualization coordinates). Below the labels, you can see the beginning of the various ties (arcs) that each novice has to other novices along with their values. That is, you can see that just as with the above matrix, actor (vertex) 1 (Ramauld) as a tie (arc) to actor 2 (Bonaventure) of strength 1, a tie to actor 3 of strength 1, a tie to actor 5 (Peter) of strength 3, and a tie to actor 15 (Amand) of strength 2. The difference, of course, is that the "0's" aren't recorded, which is why edge lists take up far less space than matrices.

```
*Vertices      18
     1 "Ramauld"    0.6494    0.2265
     2 "Bonaventure"    0.6822    0.3510
     3 "Ambrose"    0.6710    0.5425
     4 "Berthold"    0.8318    0.4507
     5 "Peter"    0.8491    0.3769
     6 "Louis"    0.7337    0.7483
     7 "Victor"    0.8500    0.6646
     8 "Winfrid"    0.4556    0.6407
     9 "John Bosco"    0.3678    0.8387
    10 "Gregory"    0.2436    0.5721
    11 "Hugh"    0.5168    0.8500
    12 "Boniface"    0.3141    0.7247
    13 "Mark"    0.3338    0.5173
    14 "Albert"    0.1624    0.6854
    15 "Amand"    0.4705    0.2758
    16 "Basil"    0.3285    0.1500
    17 "Elias"    0.1500    0.3213
    18 "Simplicius"    0.2825    0.2585
*Arcs
     1     2    1.0000
     1     3    1.0000
     1     5    3.0000
     1    15    2.0000
     2     3    1.0000
     2     5    3.0000
     2     6    2.0000
     3     2    1.0000
     3     7    3.0000
     3     8    2.0000
     4     2    1.0000
     4     3    2.0000
     4     5    3.0000
```

In what follows, we will begin by exploring UCINET, which typically works with matrices, and then we'll move to Pajek where we typically work with edge lists. R reads in various types of data. For this lab, rather than use the Sampson data, which we will use in a later lab (SNA Basics IV), we will use data collected by Stuart Koschade of the 17 individuals who participated in the first Bali bombing.[1] Koschade recorded both the ties between the individuals, as well as the strength of the tie between them.

---

[1] Stuart Koschade. 2006. "A Social Network Analysis of Jemaah Islamiyah: The Applications to Counterterrorism and Intelligence." *Studies in Conflict & Terrorism* 29:559-75.

## Part I – One-Mode Social Network Data in UCINET

*[UCINET]*
*Data>Data Editors*
*>Matrix editor*

*[UCINET Spreadsheet]*
*Transform>Fill*

1. You can enter data into UCINET UCINET's spreadsheet editor (*Data>Data Editors>Matrix editor* command).[2] You can either enter the "0's" as you go or only enter cell values that are greater or less than "0". If you choose to do the latter, after you have entered all of the values, enter the size of the matrix (i.e., the number of rows and columns) on the right side of the Spreadsheet editor and then use the *Transform>Fill* command. This will fill all empty cells with "0's".
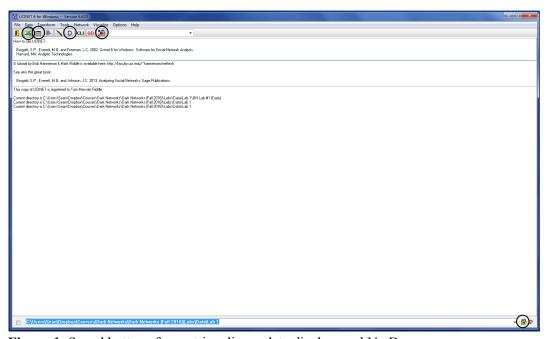


**Figure 1**: Speed buttons for matrix editors, data display, and NetDraw

*Data>Display*

2. To examine the Koschade data, select the *Display* command found under the *Data* menu, which calls up a "Display Datasets" dialog box. Click on the box to the right of "InputFn" and select the Koschade data. Alternatively, you can click on the "D" icon located on UCINET's icon menu (see Figure 1) – however, there is no dialog box with this option. The social network data you just created should now be displayed in matrix format in an output log (not shown).

3. Before proceeding any further, set the default file folder to the folder in which you have stored the Koschade data. You can do this by clicking on the file cabinet icon in the lower right of the UCINET interface (circled above), browse until you locate the

---

[2] UCINET now includes an Excel spreadsheet editor (*Data>Data Editors>Excel Matrix Editor*) that allows you to use Excel to record network data in matrix format and then save the network as a UCINET file. You can also call up the old spreadsheet editor (which I prefer) with the command *Data>Data Editors>Spreadsheet Editor (obsolete)*. Finally, there are speed buttons for both the standard spreadsheet editor and the Excel spreadsheet editor. These are circled in Figure 1.

folder, and then click "Select folder." This will save you from numerous headaches in trying to locate data and results from your analyses.

*Visualize>NetDraw*

*[NetDraw]*
*File>Open*
*>UCINET dataset>Network*

4. Included in UCINET is a graph-drawing program that works relatively seamlessly with UCINET known as NetDraw. To access NetDraw, click on the NetDraw command found under the *Visualize* menu. After NetDraw opens, open the Koschade file using the *File>Open>UCINET dataset>Network* command. Alternatively, you can click on the NetDraw icon located on the upper right of UCINET's icon menu (Figure 1 above).

*Layout>Random*

*Layout*
*>Non-metric MDS of geo*
*distances*

*Properties>Lines*
*>Size>Tie Strength*

5. NetDraw offers users several layout options that can be found under the Layout menu: Graph theoretic layout (spring embedding), Gower, Principal Components, Scaling/Ordination (metric MDS), non-metric MDS, Circle, and Random layouts. First select *Random* and then *non-metric MDS*. It is generally advisable to draw a network more than once. The more often you draw it, the closer you will arrive at an "optimal" solution. Draw the network repeatedly using non-metric MDS until the changes between drawings are minimal. Next, adjust the width of the lines, using the *Properties>Lines>Size>Tie Strength* command. Then, in the dialog box, select the network (relation) you are visualizing (in this case, the Koschade network). Your network should look similar to Figure 2.
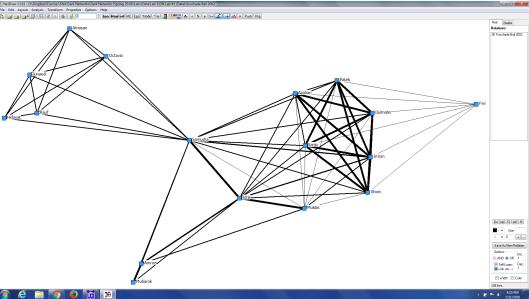


**Figure 2**: Bali I Network using non-metric MDS and Displaying Tie Strength

*Layout*
*>Graph-Theoretic Layout*

6. Next, try visualizing the network using a spring embedding algorithm, which NetDraw now calls the "graph-theoretic layout" (the lightning bolt speed button is the same thing). This should look similar to the graphs found in the Koschade article (if

you have access to it), as it should because Koschade appears to have used UCINET and NetDraw for his analysis.

*Layout
>Move/Rotate*

7. NetDraw allows users to rotate network maps (i.e., drawings) with its *Layout> Move/Rotate* command. Selecting this command brings up a dialog box (not shown) that allows users to flip, rotate, shift and vary the size (zoom) of the network drawing. It also has two "auto" commands: one re-centers the drawing, and one that adjusts the size to fit the screen. Sometimes rotating network maps help to uncover hitherto unnoticed network structures.

*[UCINET]
Data>Export
>Pajek >Network*

8. Although I've provided a Pajek file of the Koschade data, it is useful to know how to export the data from UCINET so that it can be read into Pajek. To do this, under UCINET's Data menu, choose *Export>Pajek>Network.* This will bring up a dialog box that presents you with several options. Accept UCINET's defaults, and click on the button to the right of the *[Input] – Network dataset* box. Select the data file you just created, and click "OK." Not only should this create a new file with a ".net" extension, but it should also give you the option of opening the new file in Pajek. Unless you have told UCINET where to find the Pajek executable file (under "Options), simply click "Cancel."[3]

**Part II – One-Mode Social Network Data in Pajek**

1. After opening Pajek you will see Pajek's main screen (Figure 3), which includes 6 "drop-down lists" – "Networks," "Partitions," "Vectors," "Permutations," "Cluster," and "Hierarchy" – all of which are (or at least should be) currently empty.[4] From this screen, you can open Koschade's Bali network using the *Read* command found under

*File>Network>Read*

the *File* menu or by clicking the button with an icon of a folder under the word *Networks* (circled in Figure 3). Either will open a dialog box with which you can search and select the file you just created. Be sure you select the file type" selected is "Pajek networks" (*.net), otherwise you will have difficulty locating your data.

2. When Pajek reads a network, it displays its name in the Network drop-down list, which lists all of the networks that you currently have open in Pajek. Notice that the number of actors in the network is displayed in parentheses next to the file name (this feature will be of practical use as well as helping you with a question on the final exam). *The network selected and displayed in the top drop-down list is the active*

---

[3] Note: In the first export dialog box, you can tell UCINET not to open Pajek.
[4] Pajek's main screen in Figure 3 looks different from the main screen illustrated in Everton 2012 (i.e., *Disrupting Dark Networks*) because with Pajek's developers altered Pajek's interface and menus beginning with version 3.0.

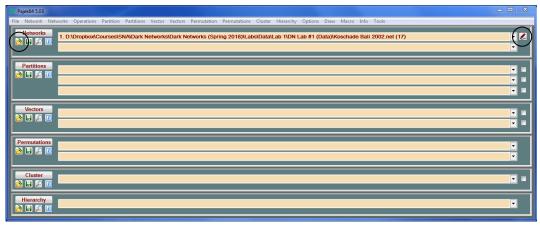*network*, which means that any operation you choose to perform on a network will use this particular network.



**Figure 3**: Pajek's Main Screen

*Draw>Network*

3. Pajek has a separate window for drawing networks, which is accessed from the *Draw >Network* menu in the Main screen or by clicking on the pencil icon to the right of the drop-down menu (also circled in Figure 6). Pajek offers several graph drawing options, some that we will cover now and some that we will cover later.[5] You can change the background, for instance, by choosing the *Colors>Background* option under the *Options* menu of the draw screen (I prefer white). Line values can be displayed by selecting the option *with Values* in the *Mark Lines* submenu of the *Options>Lines* menu. You can also display labels by selecting the "V" key while holding done the control key (Ctrl + V). Like NetDraw, you can adjust the width of the lines to reflect their value. Under the *Options* menu, select the *Lines>Different Widths* option. This will probably cause the arrows to be larger than you prefer. You can adjust the size of the arrows using Pajek's *Options>Size>of Arrows* command. Actor labels can be displayed by selecting the *Labels* option in the *Mark Vertices Using* command found in the *Options* menu. You can also display labels by selecting the "L" key while holding done the control key (Ctrl + L).

*Options>Colors >Background*

*Options>Lines >Mark Lines>with Value*

*Options>Lines >Different Widths*

*Options>Size>of Arrows*

*Options >Mark Vertices Using >Labels*

*Layout>Energy >Starting positions*

4. Pajek includes several layout commands. Two, which have been included from the beginning, are referred to as spring embedding algorithms and found under the *Layout>Energy* menu. An important thing to note is that Pajek allows researchers to choose the starting positions of the network's vertices: you can choose random and circular starting positions, or you can use the present positions of vertices as their starting positions. For now, choose the latter.[6]

---

[5] Pajek offers so many options that aside from its creators, most social network analysts (including us) have only begun to tap into all the Pajek can do.

[6] It is usually worth the time to repeat energy command a few times with given starting positions to improve a drawing. It is also advisable to experiment with both commands.

5. Before "energizing" your drawing, you first need to tell it whether the connections between actors indicate similarities/attraction/closeness or dissimilarities/repulsion/ distance in the *Options>Value of Lines* submenu of the Draw screen. When you choose *Similarities*, a higher line value between two vertices will cause them to be drawn closer together. In this case, you want to choose *Similarities* because the line values indicate attraction between the vertices. You would choose dissimilarities if a tie indicated some sort of distance between actors, such as enmity, dislike, etc.

*Options>Value of Lines*
*>Similarities*

6. The first energy command is named Kamada-Kawai after the mathematicians who developed it (*Layout>Energy>Kamada-Kawai>Free*). It produces more stable results than the other command (Fruchterman Reingold) and is preferable when drawing very sparse networks, but it is much slower and probably should be avoided when drawing networks with more than 500 vertices. With Kamada-Kawai commands *Fix first and last* and *Fix one in the middle* allow you to fix vertices that you do not want relocated. In this case, try drawing the network using the *Free* option.

*Layout>Energy*
*>Kamada-Kawai>Free*

7. The second energy command, Fruchterman Reingold, is faster and works better with large networks (*Layout>Energy>Fruchterman Reingold>2D, 3D*). It also does a better job at separating isolates from the network whereas Kamada-Kawai will sometimes draw them on isolates on top of the network. With the Fruchterman Reingold algorithm you can create two and three-dimensional drawings, something you cannot do with Kamada-Kawai, which only produces two-dimensional drawings.

*Layout>Energy*
*>Fruchterman Reingold*
*>2D, 3D*

8. Pajek allows you to rotate the network map in various ways. While looking at the drawing of the network map, first hold down the "x" key, then the "y" key, then the "z" key and finally the "s" key. Now repeat the process, except this time hold down the shift key while holding down the x, y, z, and s keys respectively. The "z" and "s" keys are generally only useful when visualizing 3D networks. Finally, select the *Spin>Spin around* command. This command calls up a dialog box that asks how many degrees to spin the drawing. Accept Pajek's default (360) and click OK.

*Spin>Spin Around*

9. A third option in the Fruchterman Reingold submenu is *Factor*; this allows you to specify the optimal distance between actors in a drawing energized by Fruchterman Reingold (but not Kamada-Kawai). A low number draws the actors closer together while a high number draws them farther apart. Experiment with both the 2D and 3D options along with the factor command.

*Layout>Energy*
*>Fruchterman Reingold*
*>Factor*

9. Pajek has implemented a couple of new drawing algorithms: Pivot MDS (Multidimensional scaling) and VOS Mapping. Explore these as well using the *Layout>Pivot MDS>Random Pivots>2D, 3D* and *Layout>VOS Mapping>2D, 3D* commands. However, the VOS Mapping algorithm only works with fully connected networks (e.g., there can't be any isolates in the network.

*Layout>Pivot MDS*
*>Random Pivots*
*>2D, 3D*

*Layout>VOS Mapping*
*>2D, 3D*