# Module 2: Data RStudio



**Goals for Module 2: Data RStudio**

~ Participants will view files outside of RStudio to be more familiar with the data structures and files before moving into RStudio.

~Participants will import three types of relational data (sociomatrix, edgelist, two-mode data) into RStudio and convert the relational data into networks.

# Data RStudio Files

The `Data RStudio Files` folder contains all of the files needed to complete the Module 2: Data RStudio. There are five files in this folder: four data files that end with the file extension ".csv" and one script file that ends with the file extension ".R". Before importing these files in RStudio, we will review their contents outside of RStudio.

The five files included in the `Data RStudio Files` folder are:

`matrix.csv`

`edgelist.csv`

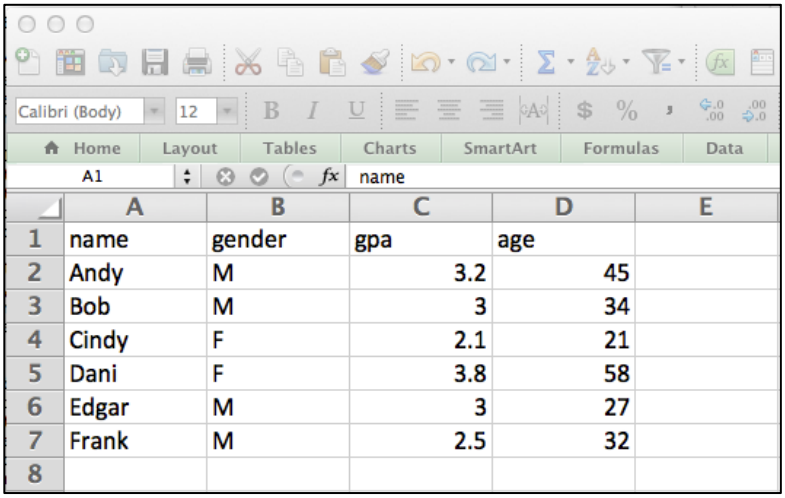
`attributes.csv`

`event.csv`

`data.R`

# CSV Files

CSV files are formats that are compatible with RStudio and easily import into RStudio software. CSV is a file extension that stands for "comma separated values." A CSV file is essentially a text file in which information is separated by commas. A CSV is a non-program specific file format that allows spreadsheet software to read text and organize it into rows and columns. Spreadsheet software, like Microsoft Excel, automatically organizes text into different cells of a table based on the location of the commas. CSV files are very versatile files that can be read easily by virtually any software program than can read text or spreadsheets on any computer. Click on the `attributes.csv` file located in the `Data RStudio Files` folder, and it should automatically open in your computer's default spreadsheet software such as Microsoft Excel or Open Office Calc. The `attributes.csv` spreadsheet contains seven rows and four columns.

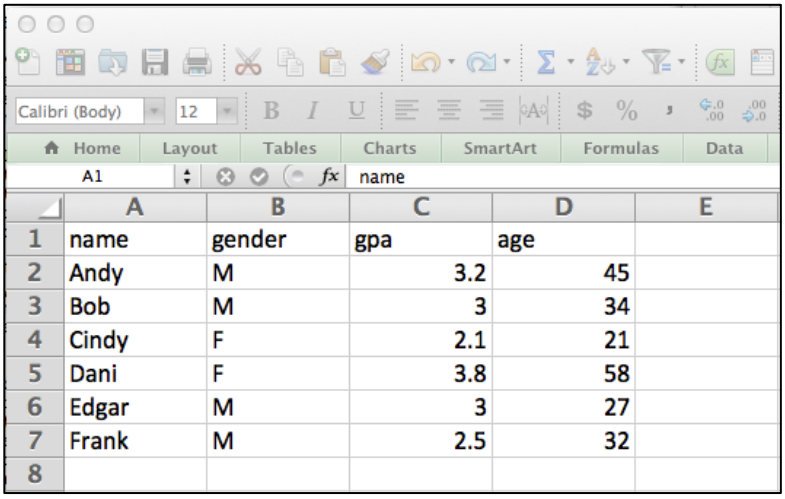



| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | name | gender | gpa | age | |
| 2 | Andy | M | 3.2 | 45 | |
| 3 | Bob | M | 3 | 34 | |
| 4 | Cindy | F | 2.1 | 21 | |
| 5 | Dani | F | 3.8 | 58 | |
| 6 | Edgar | M | 3 | 27 | |
| 7 | Frank | M | 2.5 | 32 | |
| 8 | | | | | |

To see the actual text separated by commas that underlies the CSV file, right click on `attributes.csv`, choose `Open with`, and select a text-editing program such as TextEdit on Macs or Notepad on Windows. In the text-editing program are the same seven rows seen in the spreadsheet software, but instead of four columns each row contains three commas separating the data that would otherwise belong to each separate cell across four columns.

# Attributes.csv

The **`attributes.csv`** is attribute data. In social network analysis, an attribute is a characteristic or piece of information about the nodes in the networks rather than the edges in the networks. Attributes are individual-level, meaning that they contain information on each individual node in a network. The **`attributes.csv`** contains four attributes: name, gender, grade point average, and age. Note that there is no information in the **`attributes.csv`** on how these individuals are connected. However, this spreadsheet is organized in a way that should be familiar for those with some data management experience. This spreadsheet contains individual-level information on the six people who appear in the edgelist example described below. Each row in this spreadsheet, excluding the header row, will match one dot or node in a social network. The dots or nodes in the network can vary based on the characteristics of each column. Attribute data files like this are required any time social network analysis users want to assign a piece of information to the nodes in a network.




| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | name | gender | gpa | age | |
| 2 | Andy | M | 3.2 | 45 | |
| 3 | Bob | M | 3 | 34 | |
| 4 | Cindy | F | 2.1 | 21 | |
| 5 | Dani | F | 3.8 | 58 | |
| 6 | Edgar | M | 3 | 27 | |
| 7 | Frank | M | 2.5 | 32 | |
| 8 | | | | | |

Attribute files are similar to tables used in traditional statistical analysis. Attribute files look the same for both edgelists and matrices.

# Matrix.csv

A sociomatrix is a format for organizing social network data in which cells indicate the presence (1), absence (0), or value (greater than or equal to 1) of edges. Precise ordering is required when using matrices to organize social network data. In directed networks the rows indicate the senders and the columns indicate the receivers. In undirected networks the upper and lower triangles of the sociomatrix are symmetric. The benefit of sociomatrices is that they are complete because every possible connection between individuals is accounted for.

One of the csv files in the `Data RStudio Files` folder is the `matrix.csv`. Below is a screenshot of what this csv files looks like when opened with Microsoft Excel. This sociomatrix contains data on an authentic criminal network from Montreal. The first row is a set of id numbers going across the top of the matrix. This sociomatrix does not have id numbers in the first column, but you don't need them because they are in the exact same order as those across the first row.

| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | AA | AB | AC | AD | AE | AF | AG | AH | AI | AJ | AK | AL | AM | AN | AO | AP | AQ | AR |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 |
| 2 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 |
| 3 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 7 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 8 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 9 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 10 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 12 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 13 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 15 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 16 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 17 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 18 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 19 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 20 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 21 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 22 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 23 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 24 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 25 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 26 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 27 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 28 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 29 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 30 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 31 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 32 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 33 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 34 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 35 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 36 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 37 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 38 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 39 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 40 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 41 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 42 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 43 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 44 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 45 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 46 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

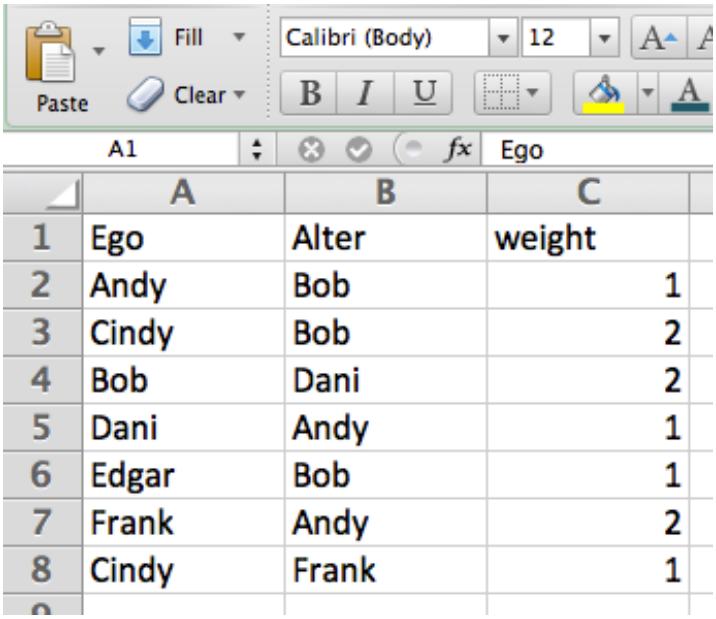
Answer the questions below using the sociomatrix above. Answers to these questions are available in the review section of this lab.

1. How many nodes are in this network?
2. Can you identify the diagonal of zeros? (Note that it starts in cell A2.)
3. Does this network appear directed or undirected?
4. Are there any empty cells in this sociomatrix?
5. Just by looking at this sociomatrix, who might be one of the most connected people in the criminal network?

# Edgelist.csv

An edgelist is a format for organizing social network data in which every row equals one edge or relationship. The order of rows does not matter, and the order of individuals in each row (who goes in column A and who goes in column B) only matters if the social network is directed. Edgelists are not good for producing isolates in networks. The benefit of edgelists is that they are easier than matrices to create and manage when the social networks are very large.

The `edgelist.csv` file contains eight rows and three columns of information about relationships in a social network. Below is a screenshot of what this CSV file looks like when opened in Microsoft Excel. Column A contains names of one person (here called "ego"), column B contains names of persons connected to the column A person (here called "alter"), and column C contains a value assigned to each edge (here called "weight"). The weight column could measure different values or number of ties between two people. Each row in this spreadsheet, excluding the header row, produces one line or edge in a social network. This network is undirected.




| | A | B | C |
|---|---|---|---|
| 1 | Ego | Alter | weight |
| 2 | Andy | Bob | 1 |
| 3 | Cindy | Bob | 2 |
| 4 | Bob | Dani | 2 |
| 5 | Dani | Andy | 1 |
| 6 | Edgar | Bob | 1 |
| 7 | Frank | Andy | 2 |
| 8 | Cindy | Frank | 1 |

Either open the `edgelist.csv` file on your own computer or use the screenshot above to answer the following questions about this edgelist. We will ignore the weight column for the time being. The answers to these questions are available in the review section of this lab.

1. How many relationships does Frank have?
2. Which node has the most edges? Which node has the fewest edges?
3. Are there any isolates in this edgelist?
4. Does the order of the edges matter? If all of row 5 switched with all of row 6, would the network still be the same?
5. Because this is an undirected network, could we switch columns A and B and get the same result?

# Event.csv

Two-mode data produce a network containing two different types of nodes. Relationships between the same types of nodes occur through shared affiliations in the other type of nodes. Projections from two-mode networks generate two one-mode networks that contain only a single type of nodes. Two-mode data are more complicated in terms of extraction, but are very useful in criminal justice applications of social network analysis. Classic arrest data containing id numbers on people who were arrested and unique identifying arrest numbers are already structured to generate two-mode networks. The `event.csv` is the final CSV file in the `Data RStudio Files` folder, and it contains two-mode or affiliation data.

| | A | B |
|---|---|---|
| 1 | person | event |
| 2 | 1 | e1 |
| 3 | 2 | e1 |
| 4 | 3 | e2 |
| 5 | 4 | e2 |
| 6 | 5 | e3 |
| 7 | 6 | e4 |
| 8 | 7 | e4 |
| 9 | 1 | e4 |
| 10 | 3 | e5 |
| 11 | 8 | e5 |

Either open the `events.csv` file on your own computer or use the screenshot of this file above to answer the following questions about two-mode data. Answers to these questions are available in the review section of this lab.

1. What are the two different types of nodes in this two-mode data set?
2. How many people are in this network?
3. How many events are in this network?
4. Was person #1 in event #5?
5. Did event #4 include person #7?
6. Which people are connected to each other through events?
7. Which events are connected to each other through people?

# Vectors, Matrices, & Data Frames


In the Module 1: Introduction RStudio you learned how to create objects that had a single assigned value (e.g., assigning an object called `a` the numerical value of 15). You also learned how to create a vector (a.k.a. an ordered list) in RStudio (e.g., assigning groceries or numbers of arrests to a single object). As you saw in the earlier sections of this lab, the CSV files that we need to import into RStudio are not single objects or vectors. They are matrices and spreadsheets (a.k.a data frames). RStudio differentiates between matrices and data frames. RStudio requires that matrices contain the same types of data—only numbers or words. Data frames, on the other hand, can contain a variety of different types of data, similar to a spreadsheet.

In the Module 1: Introduction RStudio you created a user account for Code School, an online resource that offers free tutorials on computing and programming. In this section of Module 2: Data RStudio, participants can return to Code School and complete the tutorial on vectors and matrices.

Code School tutorials do not require opening RStudio. The website includes interactive consoles that look like R where you can type your syntax, see results, and receive error messages if the syntax is incorrect.

Follow the link to the Code School Try R course:

https://www.codeschool.com/courses/try-r

Chapters in Code School are progressive, and moving on to one chapter requires the completion of the previous chapter. Participants are encouraged to complete Level 2 on vectors and Level 3 on Matrices. However, completing through Level 6 on data frames would be beneficial.

Level 2: Vectors, "Grouping values into vectors, then doing arithmetic and graphs with them."

Level 3: Matrices, "Creating and graphing two-dimensional data sets."

# Data.R in RStudio

Now we are ready to open these three different data structures in RStudio and turn them into social networks.

A. Open RStudio on your computer.

Likely there will be an icon on your desktop from the installation. If you don't have a desktop icon that looks like the graphic above, then locate RStudio among other programs on your computer.

B. Set the working directory to the **`Data RStudio Files`** folder.

1. Select the **`Session`** menu located toward the middle of the menu bar at the very top of the RStudio program.
2. Select **`Set Working Directory`**.
3. Select **`Choose Directory`**.
4. Locate the **`Lab 2 Files`** folder on your computer.
5. Click open.

C. Open the **`data.R`** script.

1. In RStudio click on the **`Open`** icon shaped like a folder located in the upper-left pane of the RStudio window.
2. Check that the folder is **`Data RStudio Files`**.
3. Select the **`data.R`** script.
4. The **`data.R`** script should open in the top-left pane. It will look like a text file with the file name at the top.
5. If the text in the script is running off the pane and requires a left to right scroll bar, select the **`Tools`** menu at the top of the RStudio window, select **`Global Options`** from **`Tools`**, select the **`Code Editing`** option from the left menu, check the box for **`Soft wrap R source files`**, and then click **`Apply`** and **`OK`**. This should set text wrap as your default in RStudio.

D. Complete Module 2: Data Lab by following the `data.R` script in RStudio. Once you have completed the script return to this document for the final review section of this Lab. Tasks in the RStudio portion of Module 2: Data include:

Setting up RStudio Session
Importing Matrix Network Data
Converting a Matrix into a Network
Importing Edgelist Network Data
Converting an Edgelist into a Network
Attaching Attributes to a Network
Importing Two-mode Event Data
Converting Event Data into Two Networks
Review

# Review of Module 2: Data RStudio

CSV is a file extension that stands for "comma separated values." A CSV file is essentially a text file in which information is separated by commas. CSV files are very versatile files that can be read easily by virtually any software program than can read text or spreadsheets on any computer.

Attribute files contain information on the characteristics of individuals in the networks. Attribute files are similar to tables used in traditional statistical analysis. Attribute files look the same for both edgelists and matrices.

A sociomatrix is a format for organizing social network data in which cells indicate the presence (1), absence (0), or value (greater than or equal to 1) of edges or relationships. Precise ordering is required when using matrices to organize social network data. The rows indicate the senders and the columns indicate the receivers. The benefit of matrices is that they are more complete because every possible connection between individuals is accounted for.

Answers to the Matrix.csv section questions:

(1) 44

(2) The diagonal of zeros begins in cell A2 and creates a diagonal line to the bottom right of the matrix ending in cell AR45.

(3) This matrix is symmetric, which means either that it is undirected or that it is directed but only containing mutual and null ties. For purposes of this lab this network is undirected.

(4) A sociomatrix is a complete dataset, so there should not be an empty cells. Zeros in cells are meaningful because they show that there is no tie between two nodes.

(5) By looking down the columns or across the rows, node #1 has the most 1s and thus is the most connected in this network.

An edgelist is another format for organizing social network data in which every row equals one edge. The order of rows does not matter, and the order of individuals in each row (who goes in column A and who goes in column B) only matters if the social network is directed. Edgelists are not very good for showing isolates in a network. The benefit of edgelists is that they are easier to create and manage when the social networks are very large.

Answers to the Edgelist.csv section questions:

(1) Frank has two relationships: one with Andy and one with Cindy.

(2) Bob has 4 edges, which is the highest in this edgelist. Edgar has only 1 edge, which is the lowest in this edgelist.

(3) Each row of this edgelist indicates an edge. This edgelist will not produce any isolates in a network. Individuals included all reside on at least a single edge or their name wouldn't make this list of edges.

(4) The order of the edges does not matter. If complete rows were switched around such as 5 and 6, it would produce the same network.

(5) In undirected networks switching column A and column B produces the same network result. This would not the case, however, if these were directed relationships.

Two-mode data and their resulting networks contain two different types of nodes. One-mode projections can be made from two-mode data that connect the nodes of a single type. Classic arrest data containing id numbers on people who were arrested and unique identifying arrest numbers are structured to generate two-mode networks.

Answers to the Events.csv section questions:

(1) people and events
(2) 8
(3) 5
(4) no
(5) yes
(6) 1 and 2 are connected through event 1
1, 6, and 7 are connected through event 4
3 and 8 are connected through event 5
3 and 4 are connected through event 4
(7) events 1 and 4 are connected through person 1
events 2 and 5 are connected through person 3

RStudio distinguishes between matrices and data frames. When all of the data are of the same type (e.g., they're all numbers or they're all text), then it is a matrix. When the data are not all of the same type (e.g., some columns are numbers, like grade point average, while other columns are text, like gender), then it is a data frame. Data frames are like spreadsheets in that they can contain a variety of different types of information. RStudio contains special functions that can be performed on matrices, so the distinction between data frames and matrices becomes important when working with the software.

Participants used the `read.csv()` command to import relational data in sociomatrix form, edgelist list form, and two-mode data form into RStudio. Participants used two commands from the igraph package to convert the matrix, edgelist, and two-mode event data into networks: `graph.adjacency()` and `graph.data.frame()`. Participants also imported an attribute file for the edgelist network and learned how to attach attribute level data to a network file using an argument within the `graph.data.frame()`command. Participants used the `V()`command in igraph to inspect node or vertex level information. Participants generated a two-mode network, and then extracted the two one-mode projections from that network.

Participants inspected their models using the basic `plot()` command with its default features. Module 3: Visualization RStudio moves through some of arguments of the `plot()` command to customize network images and look at more advanced applications of `plot()`.

Below is the list of RStudio commands used in the Module 2: Data RStudio:

```
dir()
library(igraph)
m <- read.csv("matrix.csv", header=TRUE)
m
is.matrix(m)
m <- as.matrix(m)
siren.network <- graph.adjacency(m, mode=c("undirected"))
siren.network
plot(siren.network)
e <- read.csv("edgelist.csv", header=TRUE)
e
edge.network <- graph.data.frame(e, directed=FALSE)
edge.network
plot(edge.network)
a<-read.csv("attributes.csv", header=TRUE)
a
g <- graph.data.frame(e, directed=FALSE, vertices=a)
g
V(g)$name
V(g)$gender
V(g)$gpa
V(g)$age
t <- read.csv("event.csv", header=TRUE, colClasses="character")
t
is.character(t$person)
is.numeric(t$person)
is.character(t$event)
is.numeric(t$event)
person <- unique(t$person)
event <- unique(t$event)
person
event
tf.list <- data.frame(name=c(event, person), type=c(rep(FALSE,
length(event)), rep(TRUE, length(person))), stringsAsFactors=FALSE)
tf.list
two.network <- graph.data.frame(t, directed=FALSE, vertices=tf.list)
two.network
plot(two.network)
p <- bipartite.projection(two.network)
p
event.network <- p[[1]]
person.network <- p[[2]]
event.network
person.network
plot(event.network)
plot(person.network)
```