Network Analysis With R and igraph
R Tutorial: Vectors

- Vectors can be constructed by combining their elements with the important R function `c()`.
  
  - `v1 <- c(1, 5, 11, 33)` # Numeric vector, length 4
  
  - `v2 <- c("hello","world")` # Character vector, length 2 (a vector of strings)

  - `v3 <- c(TRUE, TRUE, FALSE)` # Logical vector, same as `c(T, T, F)`
A matrix is a vector with dimensions:
- `m <- rep(1, 20)  # A vector of 20 elements, all 1`
- `dim(m) <- c(5,4)  # Dimensions set to 5 & 4, so m is now a 5x4 matrix`

Creating a matrix using `matrix()`:
- `m <- matrix(data=1, nrow=5, ncol=4)  # same matrix as above, 5x4, full of 1s`
- `m <- matrix(1,5,4)  # same matrix as above`
- `dim(m)  # What are the dimensions of m?`

Creating a matrix by combining vectors:
- `m <- cbind(1:5, 5:1, 5:9)  # Bind 3 vectors as columns, 5x3 matrix`
- `m <- rbind(1:5, 5:1, 5:9)  # Bind 3 vectors as rows, 3x5 matrix`
Create networks

- `library(igraph)` # load a package
- `detach(package:igraph)` # detach a package
- The code below generates an undirected graph with three edges.
  - The numbers are interpreted as vertex IDs, so the edges are 1→2, 2→3, 3→1.
  - `g1 <- graph( edges=c(1,2, 2,3, 3, 1), n=3, directed=F )`.

- `plot(g1)` # A simple plot of the network - we'll talk more about plots later
Igraph Object

The description of an igraph object starts with up to four letters:

- D or U, for a directed or undirected graph.
- N for a named graph (where nodes have a name attribute)
- W for a weighted graph (where edges have a weight attribute)
- B for a bipartite (two-mode) graph (where nodes have a type attribute)
Specific Graphs and Graph Models (1)

- Empty Graph
  - `eg <- make_empty_graph(40)`
  - `plot(eg, vertex.size=10, vertex.label=NA)`

- Full Graph
  - `fg <- make_full_graph(40)`
  - `plot(fg, vertex.size=10, vertex.label=NA)`

- Tree Graph
  - `tr <- make_tree(40, children = 3, mode = "undirected")`
  - `plot(tr, vertex.size=10, vertex.label=NA)`

- Simple Star Graph
  - `st <- make_star(40)`
  - `plot(st, vertex.size=10, vertex.label=NA)`
Specific Graphs and Graph Models (2)

- **Erdos-Renyi random graph model**
  - `er <- sample_gnm(n=100, m=40)`
  - `plot(er, vertex.size=6, vertex.label=NA)`

- **Watts-Strogatz small-world model**
  - `sw <- sample_smallworld(dim=2, size=10, nei=1, p=0.1)`
  - `plot(sw, vertex.size=6, vertex.label=NA, layout=layout_in_circle)`

- **Barabasi-Albert preferential attachment model for scale-free graphs**
  - `ba <- sample_pa(n=100, power=1, m=1, directed=F)`
  - `plot(ba, vertex.size=6, vertex.label=NA)`
Reading Network Data from Files: Edgelist (1)

- nodes <- read.csv("Dataset1-Media-Example-NODES.csv", header=T, as.is=T)

- links <- read.csv("Dataset1-Media-Example-EDGES.csv", header=T, as.is=T)
Reading Network Data from Files: Edgelist (2)

- `head(nodes)`
- `head(links)`

- `nrow(nodes);`
  - `length(unique(nodes$id))`

- `nrow(links)`
  - `nrow(unique(links[,c("from", "to")]))`
Reading Network Data from Files: Edgelist (3)

- links <- aggregate(links[,3], links[,3], sum)
- links <- links[order(links$from, links$to),]
- colnames(links)[4] <- "weight"
- rownames(links) <- NULL
Reading Network Data from Files: Matrix (1)

- `nodes2 <- read.csv("Dataset2-Media-User-Example-NODES.csv", header=T, as.is=T)`

- `links2 <- read.csv("Dataset2-Media-User-Example-EDGES.csv", header=T, row.names=1)`
Reading Network Data from Files: Matrix (2)

- links2 <- as.matrix(links2)
- dim(links2)
- dim(nodes2)
References:

- Some contents of the slides are adapted from:
  - http://kateto.net/networks-r-igraph