install.packages("igraph")

library(igraph)

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

nodes

links <- read.csv("Dataset/Media-Example-EDGES.csv", header=T, as.is=T)

links

Examine the data:

head(nodes) head(links)

links is an adjacency matrix for a two-mode network:

links <- as.matrix(links) dim(links) dim(nodes)

net <- graph.data.frame(links, nodes, directed=T) plot(net, edge.arrow.size=.1,vertex.label=NA) plot(net, edge.arrow.size=.1) plot(net, edge.arrow.size=.1,vertex.label=V(net)$media) as_edgelist(net, names=T)

as_adjacency_matrix(net, attr="weight") net <- simplify(net, remove.multiple = F, remove.loops = T)

plot(net, edge.arrow.size=.1,vertex.label=NA)

l<- layout.circle(net)

plot(net, layout=l,edge.arrow.size=.1,vertex.label=NA)

================ 7. Distances and paths ================

Average path length

The mean of the shortest distance between each pair of nodes in the network

(in both directions for directed graphs).

mean_distance(net, directed=F) mean_distance(net, directed=T)

We can also find the length of all shortest paths in the graph:

distances(net) # with edge weights distances(net, weights=NA) # ignore weights

We can extract the distances to a node or set of nodes we are interested in.

Here we will get the distance of every media from the New York Times.

dist.from.NYT <- distances(net, v=V(net)[media=="NY Times"], to=V(net), weights=NA) dist.from.NYT

Set colors to plot the distances:

oranges <- colorRampPalette(c("dark red", "gold")) col <- oranges(max(dist.frm.oNYT)+1) col <- col[dist.from.NYT+1]

plot(net, vertex.color=col, vertex.label=dist.from.NYT, edge.arrow.size=.1, vertex.label.color="white")

We can also find the shortest path between specific nodes.

Say here between MSNBC and the New York Post:

```
news.path <- shortest_paths(net, from = V(net)[media=="MSNBC"], to = V(net)[media=="New York Post"], output = "both") # both path nodes and edges
```

Generate edge color variable to plot the path:

```
ecol <- rep("gray80", ecount(net)) ecol[unlist(news.path$epath)] <- "orange"
```

Generate edge width variable to plot the path:

```
ew <- rep(2, ecount(net)) ew[unlist(news.path$epath)] <- 4
```

Generate node color variable to plot the path:

```
vcol <- rep("gray40", vcount(net)) vcol[unlist(news.path$vpath)] <- "gold"
```

```
plot(net, vertex.color=vcol, edge.color=ecol, vertex.label=V(net)$media,edge.width=ew, edge.arrow.mode=0)
```

Identify the edges going into or out of a vertex, for instance the WSJ.

For a single node, use 'incident()', for multiple nodes use 'incident_edges()'

```
inc.edges <- incident(net, V(net)[media=="Wall Street Journal"], mode="all")
```

Set colors to plot the selected edges.

```
ecol <- rep("gray80", ecount(net)) ecol[inc.edges] <- "orange" vcol <- rep("grey40", vcount(net)) vcol[V(net)$media=="Wall Street Journal"] <- "gold" plot(net, vertex.color=vcol, edge.color=ecol)
```

neighbors

```
neigh.nodes <- neighbors(net, V(net)[media=="Wall Street Journal"], mode="out")
```

Set colors to plot the neighbors:

```
vcol[neigh.nodes] <- "#ff9d00" plot(net, vertex.label=V(net)$media,vertex.color=vcol)
```

select edges from newspapers to online sources:

```
E(net)[ V(net)[type.label=="Newspaper"] %->% V(net)[type.label=="Online"] ]
```