

Network Visualizations

Donato Scarano

12 settembre 2018

Introduction

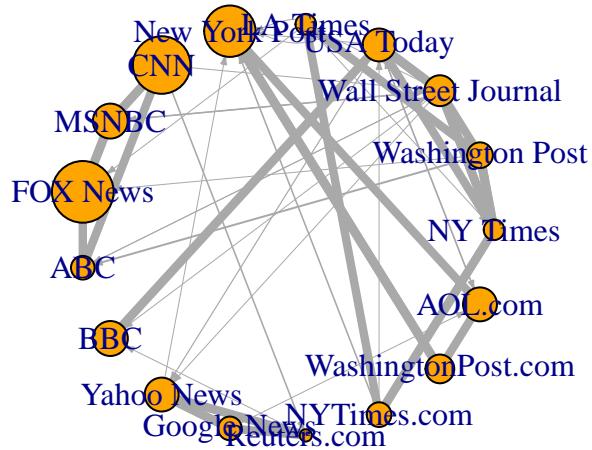
We are creating a sample of network visualization based on Airlines and Media sample network data. The data has been divided in two files one for Edges and the second one for Nodes. We load the two datasets:

```
## Parsed with column specification:  
## cols(  
##   from = col_character(),  
##   to = col_character(),  
##   type = col_character(),  
##   weight = col_integer()  
## )  
  
## Parsed with column specification:  
## cols(  
##   id = col_character(),  
##   media = col_character(),  
##   media.type = col_integer(),  
##   type.label = col_character(),  
##   audience.size = col_integer()  
## )
```

Plotting the Network

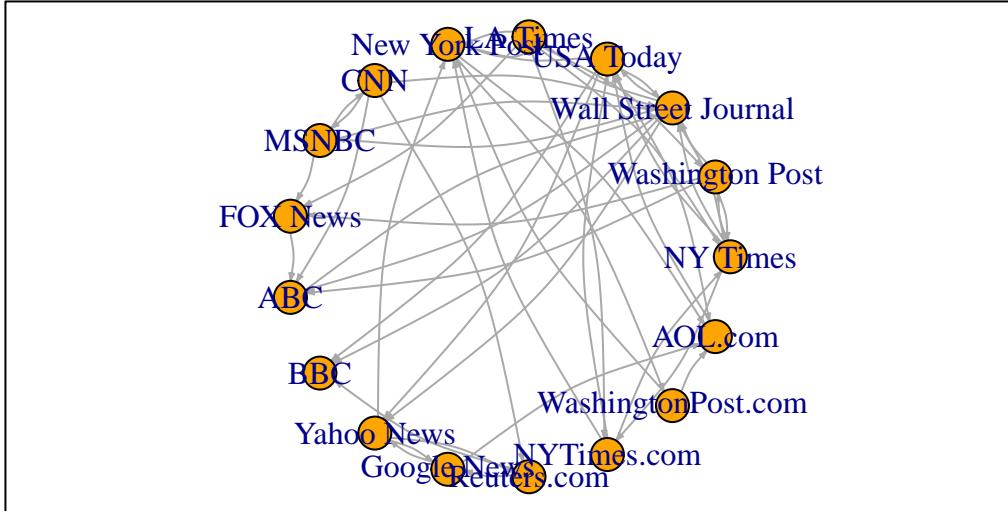
we start plotting the network using the Igraph library with size and width showing the size of the audience and weight respectively:

```
library(igraph)  
  
## Warning: package 'igraph' was built under R version 3.5.1  
##  
## Attaching package: 'igraph'  
  
## The following objects are masked from 'package:stats':  
##  
##   decompose, spectrum  
  
## The following object is masked from 'package:base':  
##  
##   union  
  
net<-graph_from_data_frame(d=DEDGES,vertices = DNODES,directed = TRUE)  
V(net)$size<-V(net)$audience.size*0.5  
E(net)$width<-E(net)$weight/5  
plot(net,edge.arrow.size=.2,edge.color="blue",vertex.color="orange",vertex.label=V(net)$media, layout=
```



Let us remove some clutter and make it simpler.

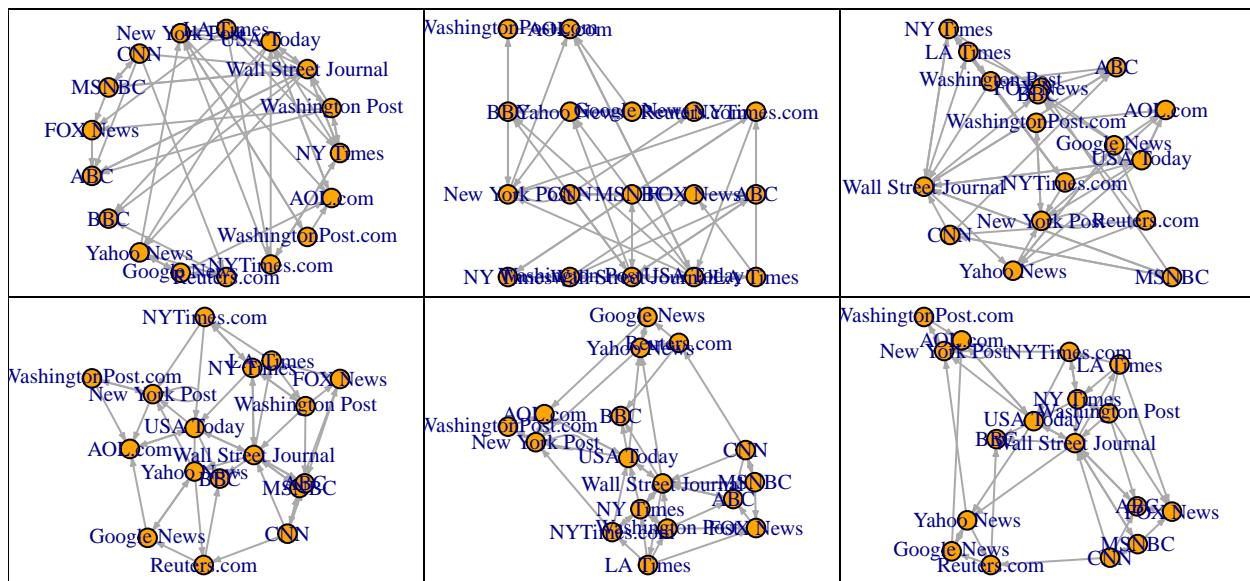
```
library(igraph)
net<-graph_from_data_frame(d=DEDGES,vertices = DNODES,directed = TRUE)
plot(net,edge.arrow.size=.2,edge.color="blue",vertex.color="orange",vertex.label=V(net)$media, layout=
```



Layout Samples

Some different layout samples illustrated (Circle, Grid, Random, Multidimensional Scaling and Fruchterman-Reingold).

```
library(igraph)
net<-graph_from_data_frame(d=DEDGES,vertices = DNODES,directed = TRUE)
par(mfrow=c(3,3),mar=c(0,0,0,0))
plot(net,edge.arrow.size=.2,edge.color="blue",vertex.color="orange",vertex.label=V(net)$media, layout="circle")
plot(net,edge.arrow.size=.2,edge.color="blue",vertex.color="orange",vertex.label=V(net)$media, layout="grid")
plot(net,edge.arrow.size=.2,edge.color="blue",vertex.color="orange",vertex.label=V(net)$media, layout="random")
plot(net,edge.arrow.size=.2,edge.color="blue",vertex.color="orange",vertex.label=V(net)$media, layout="mds")
plot(net,edge.arrow.size=.2,edge.color="blue",vertex.color="orange",vertex.label=V(net)$media, layout="fr")
plot(net,edge.arrow.size=.2,edge.color="blue",vertex.color="orange",vertex.label=V(net)$media, layout="fr")
```

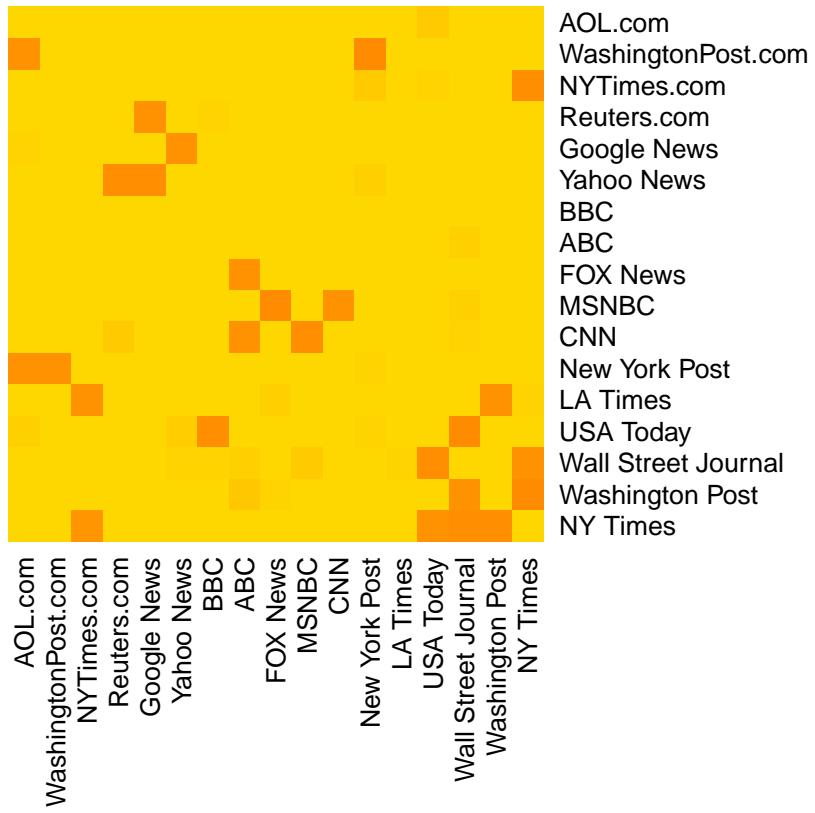


Heatmap

```

heatm <- as adjacency matrix(net, attr="weight", sparse=F)
colnames(heatm) <- V(net)$media
rownames(heatm) <- V(net)$media
palf <- colorRampPalette(c("gold", "dark orange"))
heatmap(heatm[, 17:1], Rowv = NA, Colv = NA, col = palf(100),
        scale="none", margins=c(10,10) )

```



Geographic Maps

```
## Parsed with column specification:
## cols(
##   Source = col_integer(),
##   Target = col_integer(),
##   Freq = col_integer()
## )

## Parsed with column specification:
## cols(
##   ID = col_integer(),
##   Label = col_character(),
##   Code = col_character(),
##   City = col_character(),
##   latitude = col_double(),
##   longitude = col_double(),
##   ToFly = col_integer(),
##   Visits = col_integer()
## )
```

Let us draw the geography layer and over impose the routes in this case as example we are drawing all the connections from Charlotte:

```
library("maps")
```

```
## Warning: package 'maps' was built under R version 3.5.1
```

```

library("geosphere")

## Warning: package 'geosphere' was built under R version 3.5.1
map("state")
points(x=DNODES$longitude, y=DNODES$latitude, pch=19,cex=DNODES$Visits/150, col="red")
for(i in 1:nrow(DEDGES)) {
  node1 <- DNODES [DNODES$Code=="CLT",]
  node2 <- DNODES [DNODES$ID == DEDGES[i,]$Target,]

  arc <- gcIntermediate( c(node1[1,]$longitude, node1[1,]$latitude),
                        c(node2[1,]$longitude, node2[1,]$latitude),
                        n=1000, addStartEnd=TRUE )
  edge.ind <- round(100*DEDGES[i,]$Freq / max(DEDGES$Freq))

  lines(arc)
}

```

