install.packages("igraph")

library(igraph)

g1 <- graph(edges=c(1,2, 2,3, 3, 1), n=3, directed=T) plot(g1) # A simple plot of the network

class(g1)

g1

Now with 10 vertices, and directed by default:

g2 <- graph(edges=c(1,2, 2,3, 3, 1), n=10) plot(g2)

g2

g3 <- graph(c("John", "Jim", "Jim", "Jill", "Jill", "John")) # named vertices

When the edge list has vertex names, the number of nodes is not needed

plot(g3)

g4 <- graph(c("John", "Jim", "Jim", "Jack", "Jim", "Jack", "John", "John"), isolates=c("Jesse", "Janis", "Jennifer", "Justin"))

In named graphs we can specify isolates by providing a list of their names.

plot(g4, edge.arrow.size=.1, vertex.color="gold", vertex.size=15, vertex.frame.color="gray", vertex.label.color="black", vertex.label.cex=0.8, vertex.label.dist=2, edge.curved=0.2)

E(g4)# The edges of the object

V(g4)# The vertices of the object

You can also examine the network matrix directly:

g4[]

g4[1,]

Add attributes to the network, vertices, or edges:

automatically generated when we created the network.

V(g4)$name

V(g4)$gender <- c("male", "male", "male", "male", "female", "female", "male")

Edge attribute, assign "email" to all edges

E(g4)$type <- "email"

Edge weight, setting all existing edges to 10

E(g4)$weight <- 10

Examine attributes:

edge_attr(g4)

vertex_attr(g4)

graph_attr(g4)

Another way to set attributes (you can similarly use set_edge_attr(), set_vertex_attr(), etc.):

g4 <- set_graph_attr(g4, "name", "Email Network") g4 <- set_graph_attr(g4, "something", "A thing") graph_attr_names(g4)

graph_attr(g4, "name")

graph_attr(g4)

g4 <- delete_graph_attr(g4, "something") graph_attr(g4)

plot(g4, edge.arrow.size=.1, vertex.label.color="black", vertex.label.dist=1.5, vertex.color=c("pink", "skyblue")[1+(V(g4)$gender=="male")])

V(g4)$gender=="male" returns a boolean vector, +1 converts 0–>1, 1–>2.

Then c("pink", "skyblue")[1+(V(g4)$gender=="male")]

generates a vector such that 1 is mapped to pink,

and 2 is mapped to skyblue

g4s <- simplify(g4, remove.multiple = T, remove.loops = T, edge.attr.comb=c(weight="sum", type="ignore")) plot(g4s,edge.arrow.size=.1,vertex.label.dist=1)