# Demonstrations of the igraph functions

1. ***Read the file***

```
edge1 <- read.csv("./data/Edges/edges.csv",header=F)
vertex3 <- read.csv("./data/Vertex/vertex.csv",header=F)
```

2. ***Set the format of the variable***

```
edge1[,1] = as.character(edge1[,1])
vertex3[,1] = as.character(vertex3[,1])
```

3. ***Graph build***

```
people = data.frame(id = vertex3[,1], name = vertex3[,2])
g = graph.data.frame(d=edge,direct=T,vertices=people)
```

4. ***Draw***

```
png('./data/picture.png',width=900,height=900)
set.seed(20)
plot(g, layout = layout.fruchterman.reingold, vertex.size =V(g)$size+2, vertex.color=V(g)$color
, vertex.label=V(g)$label, vertex.label.cex=1, edge.color = grey(0.5), edge.arrow.mode = "-",
edge.arrow.size=5)
dev.off()
```

Explain the variables in plot function:

*4.1 layout: Set the layout of the graph*

layout、layout.auto、layout.bipartite、layout.circle、layout.drl、layout.fruchterman.reingold、layout.fruchterman.reingold.grid、layout.graphopt、layout.grid、layout.grid.3d、layout.kamada.kawai、layout.lgl、layout.mds、layout.merge、layout.norm、layout.random、layout.reingold.tilford、layout.sphere、layout.spring、layout.star、layout.sugiyama、layout.svd

*4.2 vertex.size: set the size of different node*

```
de<-read.csv("c:/degree-info.csv",header=F)
V(g)$deg<-de[,2]
V(g)$size=2
V(g)[deg>=1]$size=4
```

```
V(g)[deg>=2]$size=6
```

*4.3 vertex.color: set the color*

```
color<-read.csv("c:/color.csv",header=F)
col<-c("red","skyblue")
V(g)$color=col[color[,1]]
```

*4.4 vertex.label:set the laber of node*

```
V(g)$label=V(g)$name
vertex.label=V(g)$label
```

*4.5 vertex.label.cex:* set the font size of the node label

*4.6 edge.color:*

```
E(g)$color="grey"
for(i in 1:length(pa3[,1])){
        E(g,path=pa3[i,])$color="red"
}
edge.color=E(g)$color
```

*4.7 Set the width of edge*

```
E(g)$width=1
```

***5. Cluster analysis***

*5.1 cluster of edge betweenness*

```
system.time(ec <- edge.betweenness.community(g))
print(modularity(ec))
plot(ec, g,vertex.size=5,vertex.label=NA)
```

*5.2 random walk*

```
system.time(wc <- walktrap.community(g))
print(modularity(wc))
#membership(wc)
plot(wc , g,vertex.size=5,vertex.label=NA)
```

*5.3 Eigenvalue*

```
system.time(lec <-leading.eigenvector.community(g))
print(modularity(lec))
plot(lec,g,vertex.size=5,vertex.label=NA)
```

*5.4 Greedy strategy*

system.time(fc <- fastgreedy.community(g))

print(modularity(fc))

plot(fc, g,vertex.size=5,vertex.label=NA)

*5.5 Muti-level clustering*

system.time(mc <- multilevel.community(g, weights=NA))

print(modularity(mc))

plot(mc, g,vertex.size=5,vertex.label=NA)

*5.6 Label propagation*

system.time(lc <- label.propagation.community(g))

print(modularity(lc))

plot(lc , g,vertex.size=5,vertex.label=NA)

***6. File output***

```
zz<-file("d:/test.txt","w")
cat(x,file=zz,sep="\n")
close(zz)
```

***7. View the variable data type and length***

```
mode(x)
length(x)
```

***8. Diameter of a graph: The diameter of a graph is the length of the longest geodesic***

diameter(graph,directed=True,unconnected=TRUE,weights=null)