igraph-project
Wojciech Mikołajczyk
May 13, 2018

Projekt R/igraph

1. Pobranie i wypakowanie datasetu

```r
library(RCurl)
```

```
## Loading required package: bitops
```

```r
datasetUrl = "http://www.sociopatterns.org/wp-content/uploads/2016/06/tij_InVS.dat_.zip"
datasetMetaUrl = "http://www.sociopatterns.org/wp-content/uploads/2016/06/metadata_InVS13.txt"
datasetZipFilename = "workplace.zip"
datasetFilename = "workplace_dataset.csv"
datasetMetaFilename = "workplace_meta.csv"

# dataset file (timestamp node1_id node2_id)
datasetBin = getBinaryURL(datasetUrl)
destFile = file(datasetZipFilename, open="wb")
writeBin(datasetBin, destFile)
close(destFile)
unzip(datasetZipFilename)
if (file.exists(datasetZipFilename))
{
    file.remove(datasetZipFilename)
}
```

```
## [1] TRUE
```

```r
file.rename("tij_InVS.dat", datasetFilename)
```

```
## [1] TRUE
```

```r
# metadata file (node_id node_attribute)
datasetMeta = getURL(datasetMetaUrl)
destMetaFile = file(datasetMetaFilename, open="w")
write(datasetMeta, destMetaFile)
close(destMetaFile)
```

2. Wczytanie grafu z datasetu

```r
library(igraph)
```

```
##
## Attaching package: 'igraph'
```

```
## The following objects are masked from 'package:stats':
##
##     decompose, spectrum
```

```
## The following object is masked from 'package:base':
##
##      union
```

```r
nodes <- read.csv(datasetMetaFilename, sep="\t", header=FALSE)
links <- read.csv(datasetFilename, sep=" ", header=FALSE, colClasses = c("NULL", NA, NA))

net <- graph_from_edgelist(as.matrix(links))
net <- set_vertex_attr(net, "department", index=nodes$V1, as.character(nodes$V2))
# set initial weight of all edges to 1
E(net)$weight <- 1
# adjust weights
net <- simplify(net, edge.attr.comb=list(weight="sum"))
# make graph undirected (meeting of 2 people is like undirected relation)
net <- as.undirected(net, mode="collapse", edge.attr.comb="sum")
# delete unconnected verticles
net <-delete.vertices(net, degree(net)==0)
```
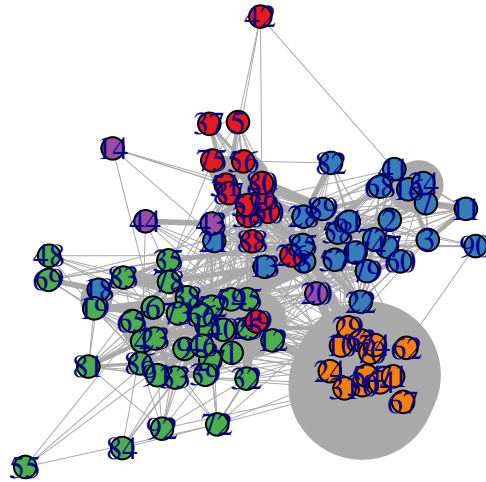
3. Wykresy

**Wykres 1 - kolor wierzchołka zależny od atrybutu, grubość krawędzi zależna od wagi**

```r
library(RColorBrewer)
numberOrDepartments <- length(unique(V(net)$department))
myColors <- brewer.pal(numberOrDepartments,"Set1")

plot(
  net,
  # weight based on edge
  edge.width=E(net)$weight/10,
  # colors
  vertex.color = myColors[as.numeric(as.factor(V(net)$department))],
  vertex.size=10
)
```

Wykres 2 - wielkość wierzchołka i grubość krawędzi zależna od pośrednictwa

```
plot(
  net,
  # edge thickness depending on edge betweenness
  edge.width=edge_betweenness(net, directed=FALSE, weights = E(net)$weight)/50,
  # vertex size based on betweenness
  vertex.size=betweenness(net, directed=FALSE, weights=E(net)$weight)/10
)
```
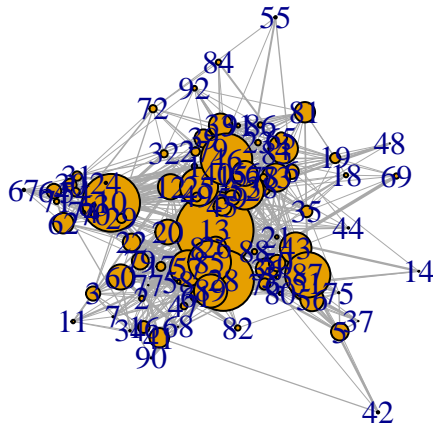
Wykres 3 - najdłuższa z najkrótszych ścieżek

```r
# shortest paths
shortestPaths <- distances(net)
longestShortestPath <- which(shortestPaths==max(shortestPaths), arr.ind = TRUE)[1,]
node1 <- longestShortestPath[1]
node2 <- longestShortestPath[2]

net.path = shortest_paths(net, from=node1, to=node2, output="both")
# edges colors
ecol <- rep("gray80", ecount(net))
ecol[unlist(net.path$epath)] <- "orange"
# edges width
ew <- rep(0.5, ecount(net))
ew[unlist(net.path$epath)] <- 5
# verticles color
vcol <- rep("gray40", vcount(net))
vcol[unlist(net.path$vpath)] <- "gold"
# verticles size
vsize <-rep(7, vcount(net))
vsize[unlist(net.path$vpath)] <- 14

plot(
  net,
```
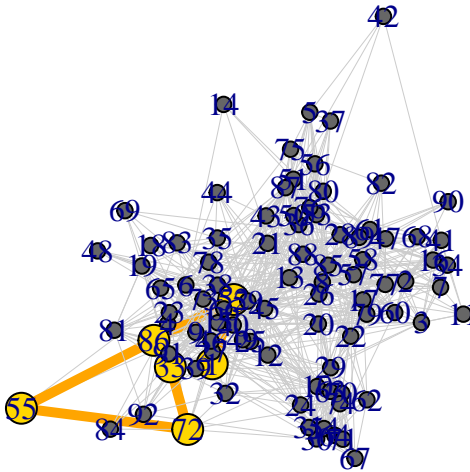
```r
  # edges - color and width
  edge.color=ecol,
  edge.width=ew,
  # verticles - color and size
  vertex.color=vcol,
  vertex.size=vsize
)
```


Wykres 4 - największa klika

```r
largest_clique <- largest_cliques(net)[1]
# color
vcol <- rep("grey80", vcount(net))
vcol[unlist(largest_clique)] <- "gold"
# size
vsize <- rep(7, vcount(net))
vsize[unlist(largest_clique)] <- 10

clique_edges = get.edge.ids(net, unlist(largest_clique))
ecol <- rep("gray80", ecount(net))
ecol[clique_edges] <- "gold"

ew <- rep(0.5, ecount(net))
ew[clique_edges] <- 5
```

```
plot(
  net,
  vertex.color=vcol,
  vertex.size=vsize,
  edge.color=ecol,
  edge.width=ew
)
```
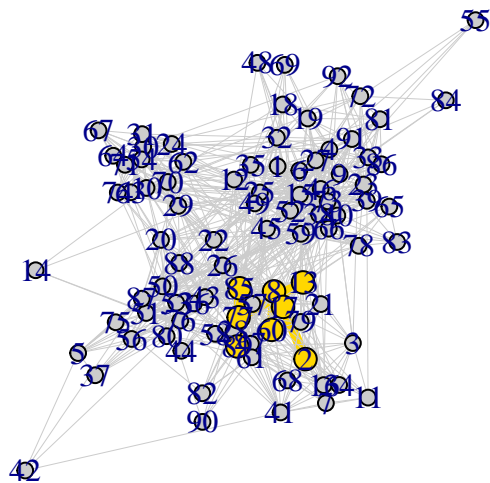

Wykres 5 - rozkład stopni wierzchołków

```
library(ggplot2)
degreeDistribution <- table(degree(net))
ggplot(as.data.frame(degreeDistribution), aes(x=Var1, y=Freq)) +
  geom_bar(stat="identity") +
  labs(title="Rozklad stopni wierzcholkow", x="Stopien wierzcholka", y="Liczba wierzcholkow") +
  theme_bw()
```

Rozklad stopni wierzcholkow
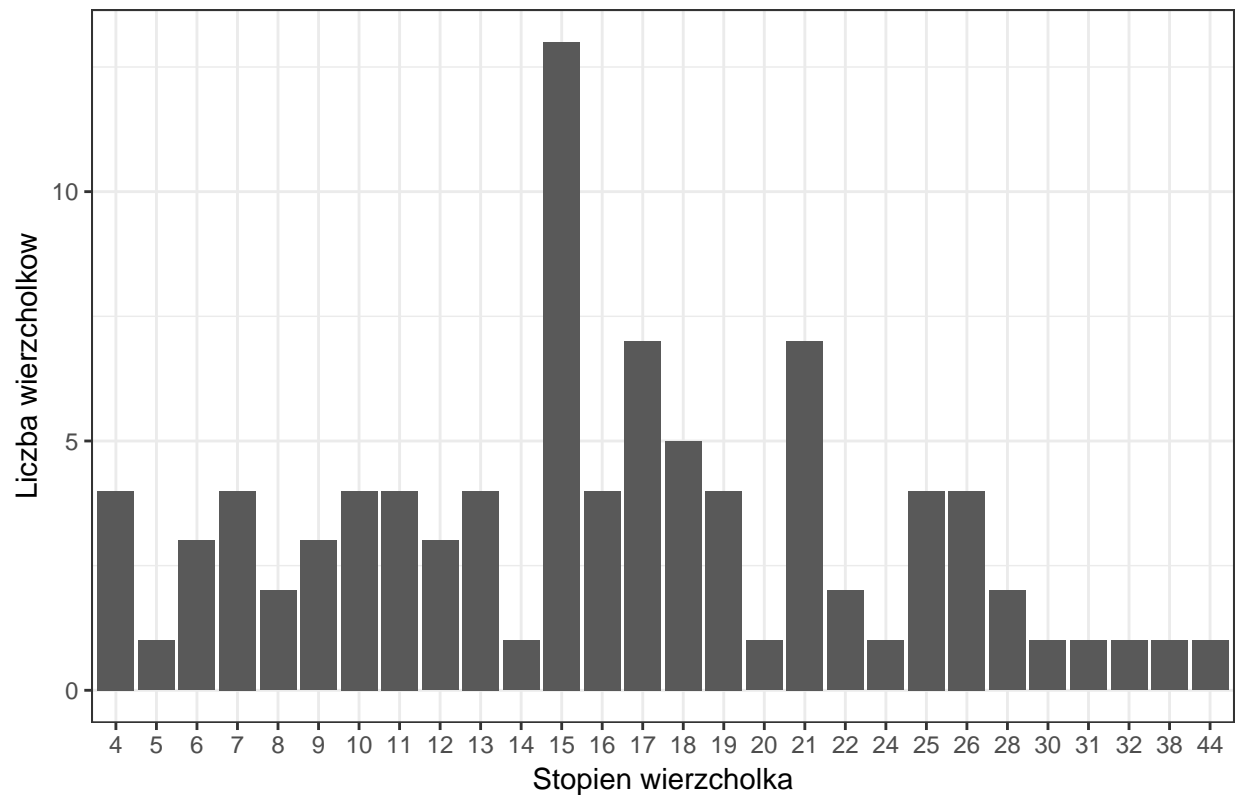
Wykres 6 - odległość z wierzchołka id=1 do pozostałych

```r
dist <- distances(net, v=1, to=V(net))

oranges <- colorRampPalette(c("gold", "dark red"))
col <- oranges(max(dist)+1)
col <- col[dist+1]

plot(net, vertex.color=col, vertex.label=dist, vertex.label.color="white")
```