

732A51 Bioinformatics

LAB 5 Bioinformatics

Hugo Morvan
William Wiik

STIMA
Department of Computer and Information Science
Linköpings universitet
2024-12-12

Contents

| | | |
|----------|-------------------|----------|
| 1 | Question 2 | 1 |
| 1.1 | Graph 1 | 1 |
| 1.2 | Graph 2 | 4 |

1 Question 2

Recreate one of the three analyses that can be found on <https://strimmerlab.github.io/software/genenet/index.html>. Document and discuss all your steps. In the analyses there is the step where you select the edges to keep. There a particular criterion is chosen for edge inclusion. Vary this criterion and explore how the resulting clusters will differ with the changes. Take one found cluster, identify the elements in it and try to find information on this cluster. Is it related to some known biological phenomena? If you do not find anything, then document your search attempts.

```
con <- gzfile("ChG-Miner_miner-chem-gene.tsv.gz", "rt")
data <- read.table(con, sep = "\t", header = FALSE)
close(con)
colnames(data) <- c("Drug", "Gene")
```

1.1 Graph 1

```
library(igraph)

# Function to create and plot a bipartite network from an edge matrix
plot_bipartite_network <- function(edge_matrix, title = "Bipartite Network") {
  # Create unique node names
  drugs <- unique(edge_matrix[,1])
  genes <- unique(edge_matrix[,2])

  # Create the graph
  g <- make_empty_graph(directed = FALSE)

  # Add all nodes
  all_nodes <- c(drugs, genes)
  g <- add_vertices(g, length(all_nodes), name = all_nodes)

  # Create type attribute for bipartite graph
  V(g)$type <- c(rep(TRUE, length(drugs)), rep(FALSE, length(genes)))

  # Add edges
  for(i in 1:nrow(edge_matrix)) {
    g <- add_edges(g,
      c(which(V(g)$name == edge_matrix[i,1]),
        which(V(g)$name == edge_matrix[i,2])))
  }

  # Set node colors based on type
  #V(g)$color <- ifelse(V(g)$type, "lightblue", "lightgreen")

  comms <- cluster_louvain(g)
  #print(length(unique(comms)))
  mems <- membership(comms)
```

```

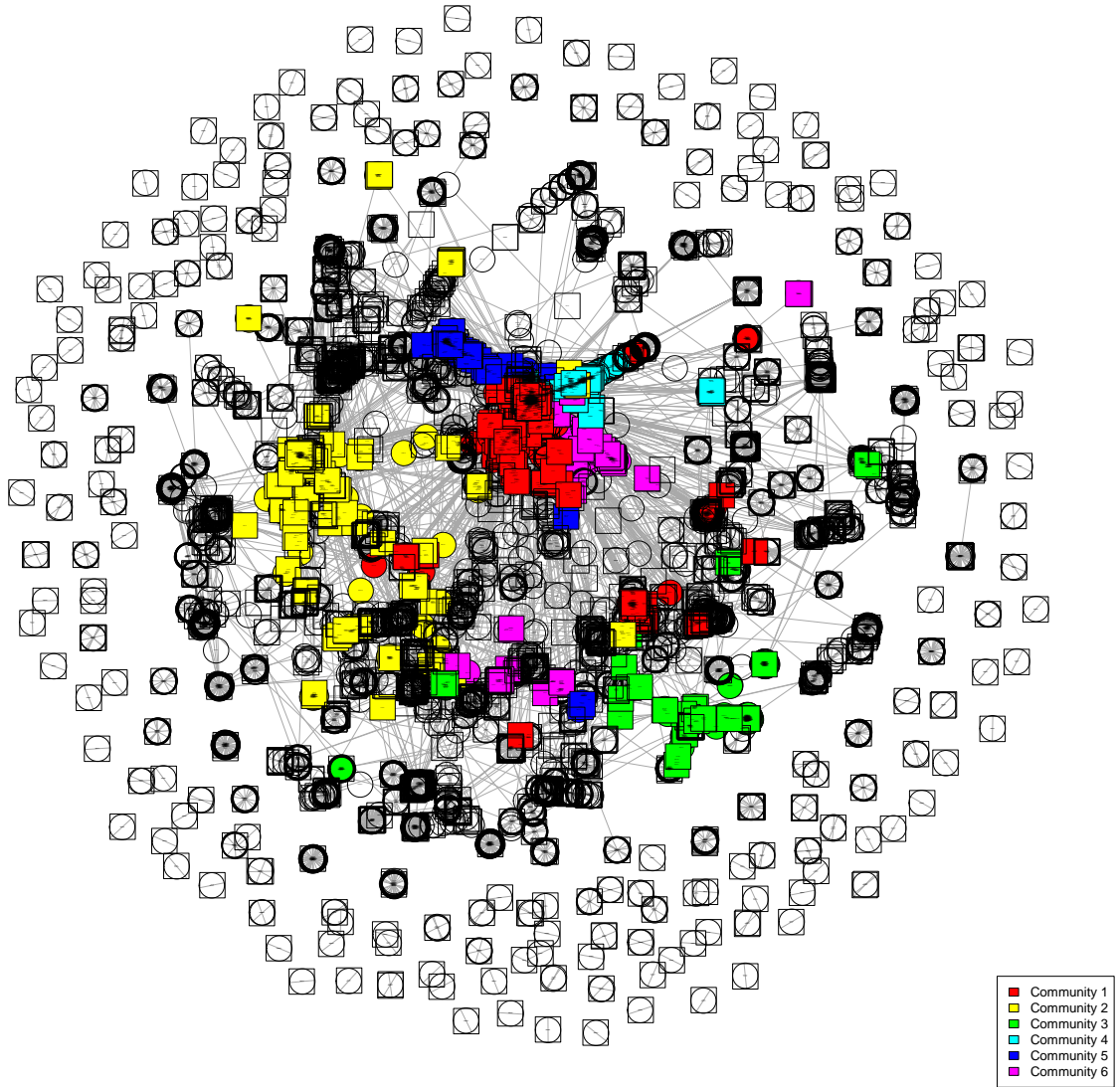
comm_col <- rainbow(length(unique(comms)))[mems]
#print(length(comm_col))
V(g)$shape <- ifelse(V(g)$type, "circle", "square")
# Plot the network
plot(g,
      vertex.label = V(g)$name,
      vertex.shape = V(g)$shape,
      vertex.color = comm_col,
      vertex.label.cex = 0.1, # Adjust label size
      vertex.size = 5,       # Node size
      vertex.label.color = "black",
      main = title#,
      #layout = layout_as_bipartite # Specific layout for bipartite graphs
)
legend("bottomright", legend = paste("Community", 1:length(unique(comms))),
      fill = unique(comm_col))
# Return the graph object for further analysis if needed
return(g)
}

# Example usage:
# Create a sample edge matrix (drug-gene connections)
edge_matrix <- data

# Plot the network
plot_bipartite_network(edge_matrix, "Drug-Gene Interaction Network")

```

Drug-Gene Interaction Network



```
## IGRAPH 904da18 UN-B 7341 15138 --
## + attr: name (v/c), type (v/l), shape (v/c)
## + edges from 904da18 (vertex names):
## [1] DB00357--P05108 DB02721--P00325 DB00773--P23219 DB07138--Q16539
## [5] DB08136--P24941 DB01242--P23975 DB01238--P08173 DB00186--P48169
## [9] DB00338--P10635 DB01151--P08913 DB01244--P05023 DB01745--P07477
## [13] DB01996--P08254 DB04800--P18031 DB08352--Q16539 DB00133--P21549
## [17] DB00163--P21266 DB00197--P10632 DB06777--P08684 DB01151--P10635
```

```
## [21] DB00356--P08684 DB01589--P34903 DB01272--P20645 DB08846--Q14534
## [25] DB01151--P33261 DB01076--P04035 DB00605--Q03181 DB08515--P49721
## [29] DB02401--P07195 DB01057--P06276 DB03286--P11217 DB08814--Q9Y233
## + ... omitted several edges
```

1.2 Graph 2

```
library(igraph)

# Function to create and plot a bipartite network from an edge matrix
plot_bipartite_network <- function(edge_matrix, title = "Bipartite Network") {
  # Create unique node names
  drugs <- unique(edge_matrix[,1])
  genes <- unique(edge_matrix[,2])

  # Create the graph
  g <- make_empty_graph(directed = FALSE)

  # Add all nodes
  all_nodes <- c(drugs, genes)
  g <- add_vertices(g, length(all_nodes), name = all_nodes)

  # Create type attribute for bipartite graph
  V(g)$type <- c(rep(TRUE, length(drugs)), rep(FALSE, length(genes)))

  # Add edges
  for(i in 1:nrow(edge_matrix)) {
    g <- add_edges(g,
      c(which(V(g)$name == edge_matrix[i,1]),
        which(V(g)$name == edge_matrix[i,2])))
  }

  # Set node colors based on type
  #V(g)$color <- ifelse(V(g)$type, "lightblue", "lightgreen")

  comms <- cluster_louvain(g)
  #print(length(unique(comms)))
  mems <- membership(comms)
  comm_col <- rainbow(length(unique(comms)))[mems]
  #print(length(comm_col))
  # Plot the network

  plot(g,
    vertex.label = V(g)$name,
    vertex.color = comm_col,
    vertex.label.cex = 0.1, # Adjust label size
    vertex.size = 5,       # Node size
```

```

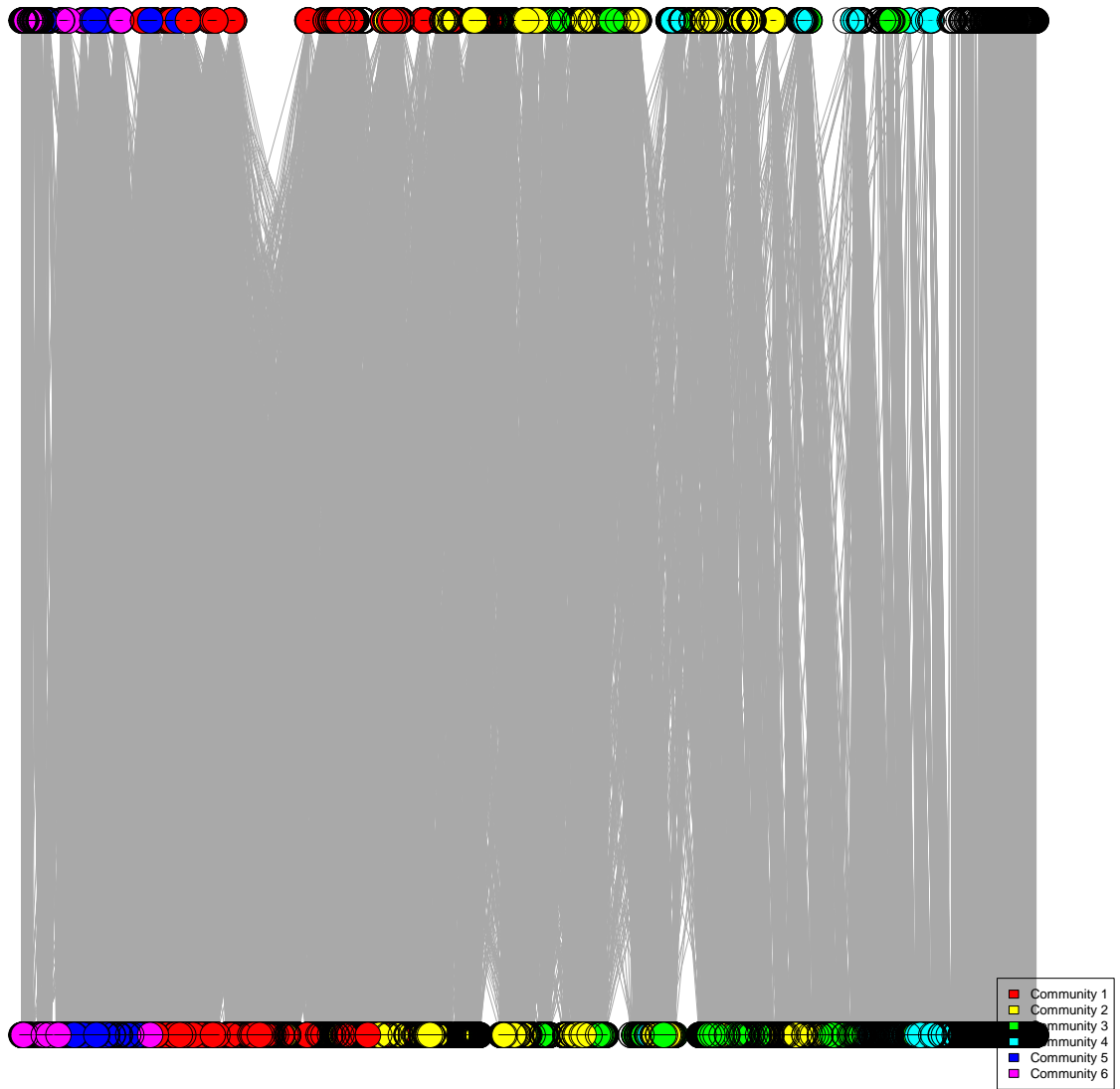
    vertex.label.color = "black",
    main = title,
    layout = layout_as_bipartite # Specific layout for bipartite graphs
  )
  legend("bottomright", legend = paste("Community", 1:length(unique(comms))),
        fill = unique(comm_col))
  # Return the graph object for further analysis if needed
  return(g)
}

# Example usage:
# Create a sample edge matrix (drug-gene connections)
edge_matrix <- data

# Plot the network
plot_bipartite_network(edge_matrix, "Drug-Gene Interaction Network")

```

Drug-Gene Interaction Network



```
## IGRAPH bbb49e4 UN-B 7341 15138 --
## + attr: name (v/c), type (v/l)
## + edges from bbb49e4 (vertex names):
## [1] DB00357--P05108 DB02721--P00325 DB00773--P23219 DB07138--Q16539
## [5] DB08136--P24941 DB01242--P23975 DB01238--P08173 DB00186--P48169
## [9] DB00338--P10635 DB01151--P08913 DB01244--P05023 DB01745--P07477
## [13] DB01996--P08254 DB04800--P18031 DB08352--Q16539 DB00133--P21549
## [17] DB00163--P21266 DB00197--P10632 DB06777--P08684 DB01151--P10635
```



```
## [21] DB00356--P08684 DB01589--P34903 DB01272--P20645 DB08846--Q14534
## [25] DB01151--P33261 DB01076--P04035 DB00605--Q03181 DB08515--P49721
## [29] DB02401--P07195 DB01057--P06276 DB03286--P11217 DB08814--Q9Y233
## + ... omitted several edges
```