Project 5

Logan Bolton

2025-02-23

Acknowledgement: This code was created through the repurposing of code found in the lecture notes and through collaboration with ChatGPT-4o by OpenAI. AI tools were very helpful for me while fixing errors and determining the correct syntax to plot graphs.

```
url1 <- "https://raw.githubusercontent.com/JeffreyAlanSmith/Integrated_Network_Science/master/data/affil
url2 <- "https://raw.githubusercontent.com/JeffreyAlanSmith/Integrated_Network_Science/master/data/affil

affiliations96 <- read.delim(file = url1, check.names = FALSE)
affiliations97 <- read.delim(file = url2, check.names = FALSE)
dim(affiliations96)
```

```
## [1] 1295   91
```

1 - 1996 Dataset

a - Which student clubs serve to integrate the school and which are more peripheral?

The most core clubs are the Spanish Club, the Pep Club and NHS. The Choir barbershop quartet (4 men), the Cross Country girls 8th grade team and the boys Swim & Dive Team are the most peripheral clubs.

```
G96 <- graph_from_incidence_matrix(as.matrix(affiliations96))
```

```
## Warning: `graph_from_incidence_matrix()` was deprecated in igraph 1.6.0.
## i Please use `graph_from_biadjacency_matrix()` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
# Split nodes into types by rows versus columns
# V(G96)$type # TRUE = club, FALSE = student,
clubs <- V(G96)[V(G96)$type == TRUE]

club_degrees <- degree(G96, v = clubs)
# Identify top & bottom
cat("top clubs\n")
```

```
## top clubs
```

```
head(sort(club_degrees, decreasing = TRUE), 3)
```

```
## Spanish Club     Pep Club          NHS
##          199          157          124
```

```
cat("Peripheral clubs\n")
```

```
## Peripheral clubs
```

```r
head(sort(club_degrees, decreasing = FALSE), 3)
```

```
## Choir, barbershop quartet (4 men)          Cross Country, girls 8th
##                                 3                                 3
##             Swim & Dive Team, boys
##                                 3
```

b - Which student clubs tend to share members at high rates?

The following pairs of clubs share the most amount of members:

- Pep Club & Spanish Club
- Debate club & Forensics (National Forensics League)
- Forensics club & Forensics (National Forensics League)
  - "Forensics club" and "Forensics (National Forensics League)" are listed as separate clubs in the dataset, despite having similar names

```r
projections <- bipartite_projection(G96)
club_graph <- projections$proj2  # This is the club-club graph

# Weight = number of shared members
E(club_graph)$weight
```

```
##   [1]  9  2  6  6  4  4  1  2  5  4  5  1  1  5  2  1  2  4  2 10  4  4  2  1
##  [25]  2  2  1  3  3  1  2  4  4  5  2  3  2  2  2  5  1  2  1  3  1  3  1  8
##  [49]  5  3  2  1  2  7  8  2  1  2  3  1  1  1  1  1  2  1  5  1  1  1  2  1
##  [73]  2  1  1  4  1  2  1  2  1  1  2  1  1  1  1  1  2  1  2  2  1  2  1  1
##  [97]  1  3  2  1  1  1  1  1  1  1  1 11 13  6  1  7  4 10  4  2  1  1  5  4
## [121]  1  4  2  3  4  3  1  1  2  2  1  1  2  1  1  5  1  1  1  1  1  1  1  2
## [145]  1  1  1  1  1  1  3  2  1  2  2  3  1  1  1  4  1  1  2  1  1  2  2  1
## [169]  1  1  1  1  1  1  3  2  6  1  3  2  4  2  1  1  1  1  1  1  1  2  1  2
## [193]  4  1  2  2  1  1  1  1  1  1  1  1  1  1  3 10  5  8  1  1  1  2  2  4
## [217]  3  1  1  4  3  3  1  1  1  1  2  1  2  2  4  4  1  1  2  1  1  1  1  4
## [241]  2  1  1  1  1  1  1  2  2  1  1  1  1  2  1  2  1  1  1  1  1  9  9 10
## [265]  3  3  1  1  1  1  1  2  1  1  1  5  3  3  4  5  4  1  1  2  1  2  2  1
## [289]  1  2  1  1  1  2  2  4  2  2  2  3  1  2  6  3  3  4  3  1  1  1  1  1
## [313]  2  2  2  1  1  1  3  2  1  5  4  5  2  5  6  4  2  1  3  1  1  3  2  2
## [337]  1  1  1  1  2  1  1  1  1  1  8  1  2  1  2  1  2  1  1  8  3  5  2  1
## [361]  3  1  1  1  1  4  3  6 10  1  3  1  2  2  1  2  1  1  1  2  5  2  1  1
## [385]  1  2  1  1  1  1  1  1  1  2 10  1  1  1  1  1  3  1  1  1  1  1  2  1
## [409]  1  1  2  1  1  1  2  2  1  1  2  2  1  2  1  1  1  1  2  2  1  1  1  1
## [433] 19  7 10  1  5  3 19  3  5  4 16  4  8  5 12  3  4  2  3  7  7  2  2  5
## [457]  4  5  3  1  1  1  4  3  2  8  3  1  1  2  3  3  1  1  1  3  2  1  2  1
## [481]  1  2  1  3  3  9  6  3  3  4  3  3  4  2  1  3  2  1  1  1  1  1  1  1
## [505]  1  1  1  1  1 11 21 18  3  5  2  3  3  1  3  1  3  4  1  4  3  2  1  3
## [529]  1  2  2  3  2  3  5  3  1  1  1  2  2  2  1  2  1  2  1  4  6 19  2  3
## [553]  2  1  2  5  5  1  1  1  4  2  2  2  1  1  1  1  1  1 16  8 11  3  9  5
## [577] 18  3  2  3  4  8  7  7  6  2  2  2  2  2  1  1  1  1  3  2  1  1  2  1
## [601]  1  2  2  3  3  2  2  2  7  2  1  2  3  3  3  4  1  1  3  1  1  1  2  2
## [625]  2  2  1  1  1  1  4  4  2  1  7  1  1  1  1  4  3  1  1  1  1  2  7  2
## [649]  1  1  1  2  1  2  1  2  1  1  3  1  4  1  5  3  3  1  3  3  2  1  1  2
## [673]  1  1  1  1  1  1  1 46  3 15 26  4  1 10 10  4  5  3  5  5  4  4  2  3
## [697]  3  5  9  1  1  1  1  1  1  2  1  1  2  1  2  1  2  2  3 19  1  2  2  2
## [721]  6  1  1  1  1  1  4  1  2  7  3  2  2  1  1 32  8  3 11  3  1 15 23 14
```

```
##  [745] 14  9 31  8  6  2  3  4 11  2  5  1  3  7  9  1  4  2  3  2  6  1  7  3
##  [769]  1  1  3  1  3  2  2  1  1  2  2  1  2  2  1  1 12  2  6  3  2  5  8  8
##  [793]  6  1  2  1  1  4  3  1  1  2  1  1  1 11  4  7  3  2  3  2  2  2  3  2
##  [817]  3  4  1  2  1  1  1  1 42  4 11  1 13 14  6  9  5  5 10  1  1  7  4  3
##  [841]  1  2  9  4  6  7  1  1  1  1  1  1  1  1  1  1  3  2  5 10 16 13 17  8
##  [865]  8  9  6  5  1 15 10  6  5  5  1  2  6  1  3  2  1  1  2  1  1  1  2  1
##  [889]  1  1  1  2  1  2  2  1  1 12 15  6 30 23 11  4 12  1  2  1  5  4  5  1
##  [913]  4  2  1  1  7  1  1  3  2  2  1  1  1  3  1  1 14  7  1  6 15  3  3  2
##  [937]  2  1  2  3  1  1  1  1  1  2  1  1  1 21  8  2  3  4  5  2  1  3  1  2
##  [961]  3  1  2  1  6  3  1  1  1  1  1 15  2  6  4  1  1  2  3  3  2  1  1  2
##  [985]  2  1  2  1  1  5  2  7  2  7  9  3  3  3  2  6  5  2  2  3  4  2  4  2
## [1009]  2  2  1  1  5  1  2  2  2  1  1  1  5  1  2  2  2  1  1  1  1  1  1  1
## [1033]  3  1  3  1  1  1  1  1  2  9  2  1  3  1  6  1  9  4  3  2  1  1  1  1
## [1057]  1 17  9  5  3  6  2  2  9  2  2  1  2  1  4  2  6  3  1  4  1  1  1  1
## [1081]  1 14  1 30  3  8 16 11  2  5  3  3 21 11  6  8  6  2  3  2  1  2  1  2
## [1105]  1  1 39  4 10  6  4  3  4  2 36  5  6  3  4  2  2  6  4  2  1  2  5  2
## [1129]  4  2  2  1  5  1  1  3  1  1  1 29 25 20  9 10 12  8 13  3 10  2 20  1
## [1153]  2  3  1  1  3  4  4  5  1  1  2  5  9  2  1  2  1  1  1  1 11  2  2  7
## [1177]  2  1  3  1  1  1  1 14 24  2  4 10  3  3  5  1  2  5  8  1  1  1  1  1
## [1201]  4  1  1  1  1 17  4  4  4  3  1  2  1  1  2  4  2  1  2  1  1  1  2  1
## [1225]  6 47  2  3 13  7  6  6  5  9  3  2  4 13  1  3  1  7  2  1  2  1  1  1
## [1249]  2  1  1  2  2  1  1  2  2  1  4  1  3  1  2  1  3  1  2  1  2  1  1  5
## [1273]  2  1  2  1  1  1  1  2  2  2  2  1  4  1  1  3  2  6  2  3  8  4  1  4
## [1297]  8  4  1  2  1  4  1 27  8  3  3  5  1  4  1  1  2  7  2  3  2  1  3  1
## [1321]  1  3  1  1  4  1  1  1  1  1  2  1  1  2  1  1  1  1  1  1  1 12  2  6
## [1345]  2  1  1  2  1  4  4  2  1  7  1  1  1  1  1  1  1
```

```r
# Top pairs with highest shared members
top_pairs <- as_data_frame(club_graph, what = "edges")
top_pairs <- top_pairs[order(-top_pairs$weight), ]
head(top_pairs, 3)
```

```
##          from                                    to weight
## 1226  Pep Club                          Spanish Club     47
## 680     Debate Forensics (National Forensics League)     46
## 825  Forensics Forensics (National Forensics League)     42
```

c - What is the shared feature, or theme, that brings these clubs together in a cluster?

When analyzing the clusters based off the walktrap algorithm, the majority of the clubs in the clusters share the following common traits:

- Cluster 1
  - Girls 8th grade sports teams
- Cluster 2
  - High school boys sports teams
- Cluster 3
  - Boys 8th grade sports teams
- Cluster 4
  - Majority female academic clubs and girls sports
- Cluster 5
  - Academic organizations and girls sports
- Cluster 6
  - 9th grade girls sports

- Cluster 7
  - Music clubs

I tried several different clustering methods and the walktrap algorithm gave the most coherent clusters.

```r
# club_comm <- cluster_louvain(club_graph)
# club_comm <- cluster_edge_betweenness(club_graph)
club_comm <- cluster_walktrap(club_graph)

club_membership <- membership(club_comm)
club_names <- names(club_membership)

# Combine club names and cluster numbers
club_clusters <- data.frame(
  Club = club_names,
  Cluster = club_membership
)

# View the clubs in each cluster
split(club_clusters$Club, club_clusters$Cluster)
```

```
## $`1`
## [1] "Basketball, girls 8th"    "Cross Country, girls 8th"
## [3] "Track, girls 8th"         "Volleyball, 8th"
##
## $`2`
##  [1] "Band, Jazz"                  "Band, Marching (Symphonic)"
##  [3] "Baseball, JV (10th)"         "Baseball, V"
##  [5] "Basketball, boys 9th"        "Basketball, boys JV"
##  [7] "Basketball, boys V"          "Cross Country, boys V"
##  [9] "Football, 9th"               "Football, V"
## [11] "Golf, boys V"                "Hispanic Club"
## [13] "Orchestra, Symphonic"        "Soccer, V"
## [15] "Swim & Dive Team, boys"      "Tennis, boys V"
## [17] "Track, boys V"               "Wrestling, V"
##
## $`3`
## [1] "Band, 8th"                "Basketball, boys 8th"
## [3] "Cross Country, boys 8th" "Football, 8th"
## [5] "Track, boys 8th"          "Wrestling, 8th"
##
## $`4`
##  [1] "Academic decathalon"
##  [2] "Art Club"
##  [3] "Cheerleaders, JV"
##  [4] "Cheerleaders, Spirit Squad"
##  [5] "Cheerleaders, V"
##  [6] "Choir, women's ensemble"
##  [7] "Debate"
##  [8] "Drill Team"
##  [9] "Drunk Driving"
## [10] "Forensics"
## [11] "Forensics (National Forensics League)"
## [12] "German Club"
## [13] "Key Club"
```

```
## [14] "Latin Club"
## [15] "PEER"
## [16] "Pep Club"
## [17] "Pep Club Officers"
## [18] "Softball, JV (10th)"
## [19] "Softball, V"
## [20] "STUCO"
## [21] "Teachers of Tomorrow"
## [22] "Tennis girls V"
## [23] "Track, girls V"
## [24] "Volleyball, JV"
## [25] "Yearbook Contributors"
##
## $`5`
##  [1] "Asian Club"                     "Basketball, girls JV"
##  [3] "Basketball, girls V"            "Chess Club"
##  [5] "Choir, a capella"               "Choir, barbershop quartet (4 men)"
##  [7] "Choir, chamber singers"         "Choir, vocal ensemble (4 women)"
##  [9] "Close-up"                       "Cross Country, girls V"
## [11] "Drunk Driving Officers"         "French Club (high)"
## [13] "French NHS"                     "Full IB Diploma Students (12th)"
## [15] "German NHS"                     "Internships"
## [17] "Junior Class Board"             "Newspaper Staff"
## [19] "NHS"                            "Quiz-Bowl (all)"
## [21] "Science Olympiad"               "Spanish Club (high)"
## [23] "Spanish NHS"                    "Theatre Productions"
## [25] "Thespian Society (ITS)"         "Volleyball, V"
## [27] "Yearbook Editors"
##
## $`6`
## [1] "Basketball, girls 9th"   "Cheerleaders, 9th"
## [3] "Choir, concert"          "French Club (low)"
## [5] "Orchestra, Full Concert" "Spanish Club"
## [7] "Swim & Dive Team, girls" "Volleyball, 9th"
##
## $`7`
## [1] "Cheerleaders, 8th" "Choir, treble"     "Orchestra, 8th"
```

2 - 1997 Dataset

a - What is the order, size, and density of G?

```r
G <- graph_from_incidence_matrix(as.matrix(affiliations97))

order_G <- gorder(G)
size_G <- gsize(G)
density_G <- edge_density(G)

cat("Order (number of vertices):", order_G, "\n")
```

```
## Order (number of vertices): 1386
```

```r
cat("Size (number of edges):", size_G, "\n")
```

```
## Size (number of edges): 2641
```

```r
cat("Density:", density_G, "\n")
```

```
## Density: 0.002751601
```

b - Is the network G connected? If not, what fraction of vertices belong to the largest connected component? If the network is not connected, consider only the largest component H for the remaining questions.

```r
is_connected(G)
```

```
## [1] FALSE
```

```r
cat("The graph is NOT connected\n\n")
```

```
## The graph is NOT connected
```

```r
comp <- components(G)

# Size of largest component
largest_comp_size <- max(comp$csize)
fraction_largest <- largest_comp_size / gorder(G)

# Create subgraph H
H <- induced_subgraph(G, which(comp$membership == which.max(comp$csize)))

cat("Percent of vertices belonging to largest component: ", fraction_largest, "\n")
```

```
## Percent of vertices belonging to largest component:  0.6789322
```

c - What is the average path length of H?

```r
average_path_length_H <- average.path.length(H, directed = FALSE)
```

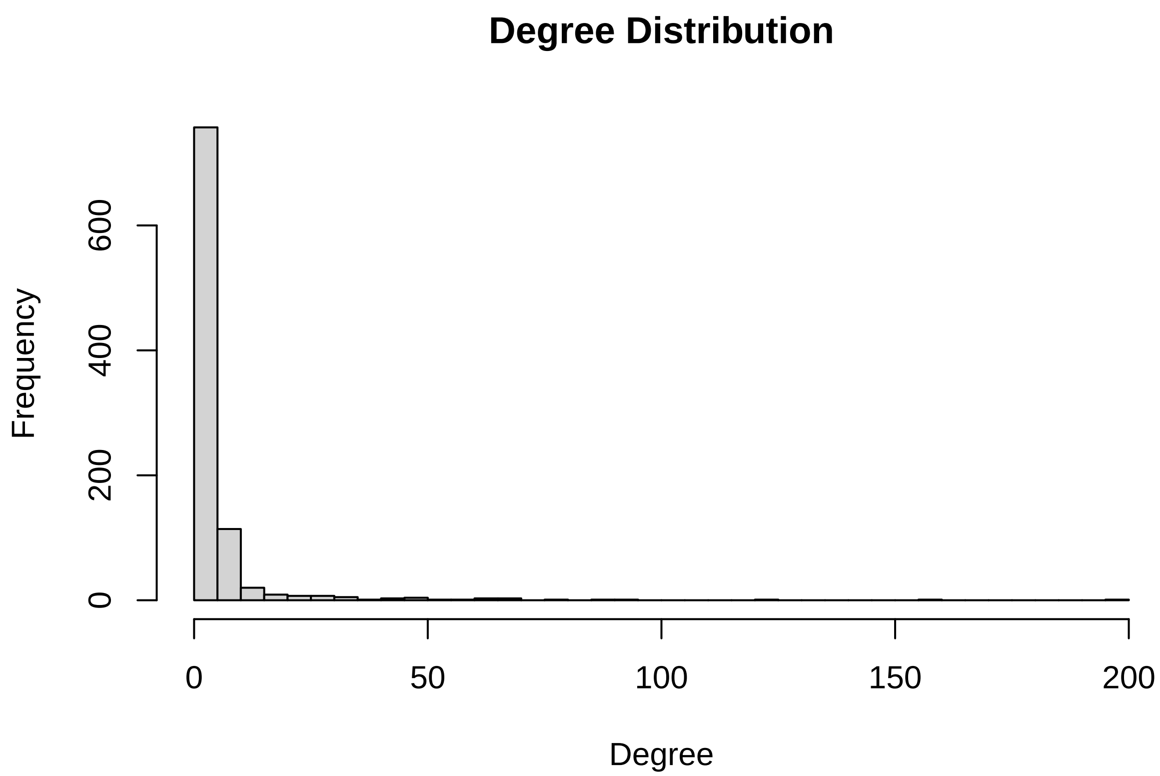
```
## Warning: `average.path.length()` was deprecated in igraph 2.0.0.
## i Please use `mean_distance()` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```r
cat("Average path length: ", average_path_length_H)
```

```
## Average path length:  3.738472
```

d - Is H scale-free? Provide statistical evidence (e.g., by examining the degree distribution and fitting a power-law distribution)

```r
deg <- degree(H)
hist(deg, breaks=50, main="Degree Distribution", xlab="Degree")
```

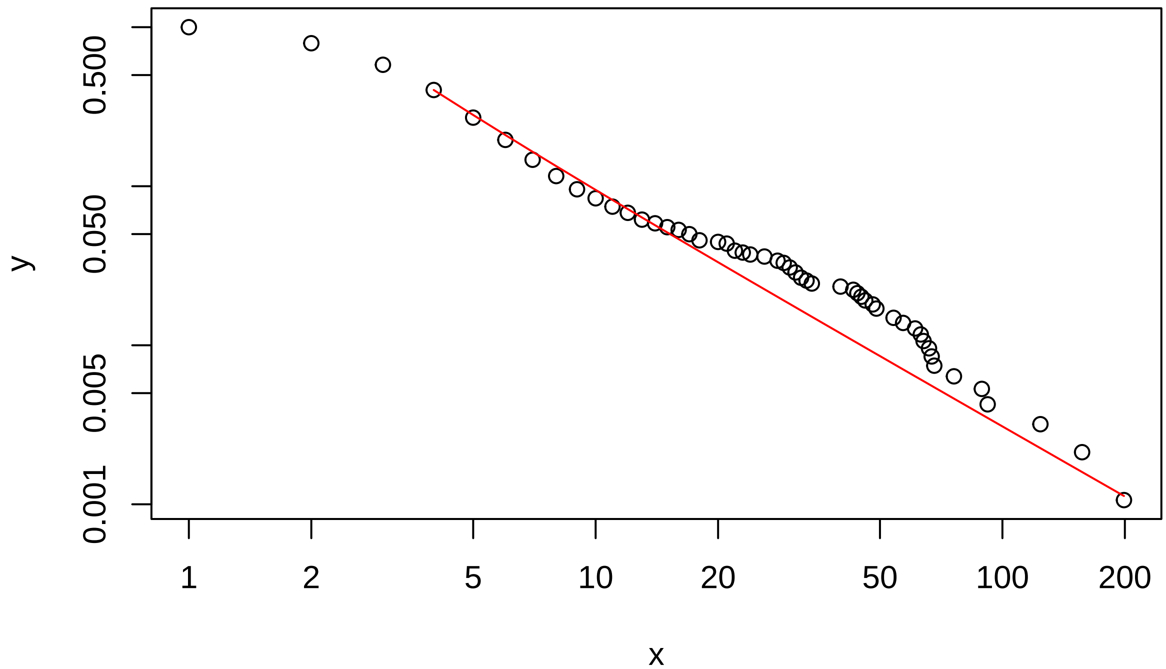



```
# Fit power law
fit <- fit_power_law(deg)
fit
```

```
## $continuous
## [1] FALSE
##
## $alpha
## [1] 2.458653
##
## $xmin
## [1] 4
##
## $logLik
## [1] -977.9973
##
## $KS.stat
## [1] 0.04464729
```

e - What is the fraction of edges that are attached to the top 10% of high-degree vertices?

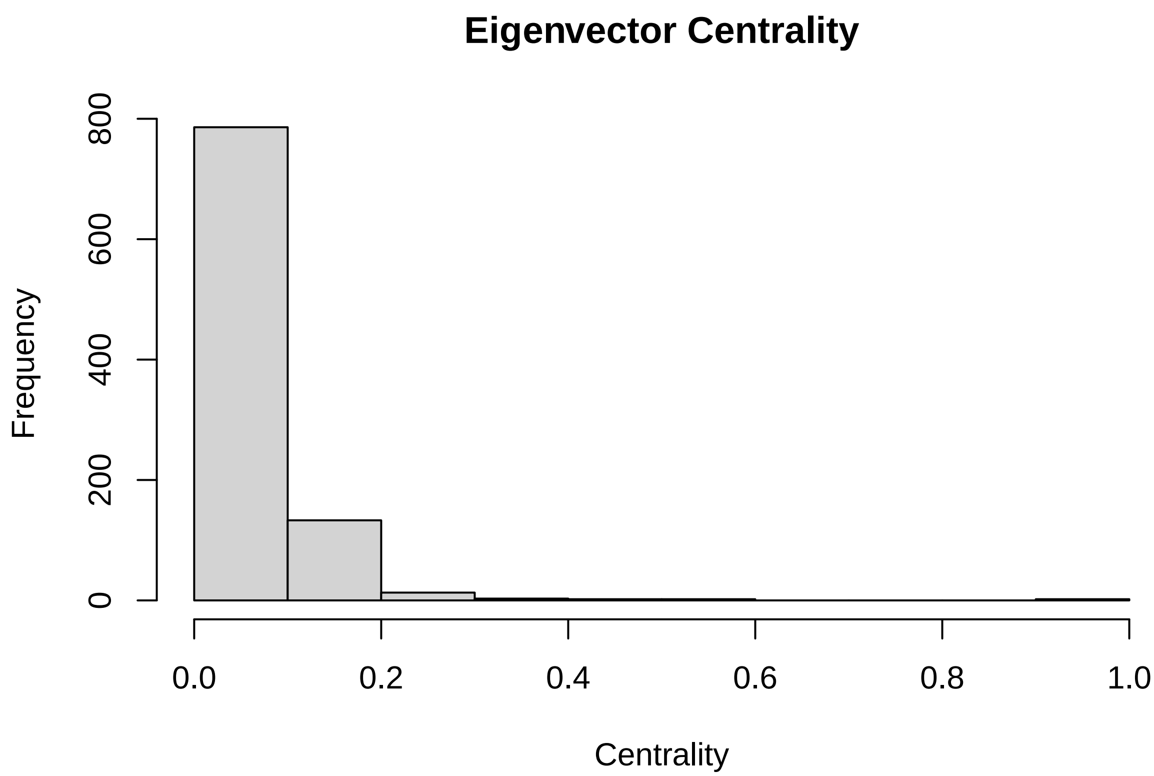
```
m <- displ$new(deg)
est <- estimate_xmin(m)
m$setXmin(est)
plot(m)
lines(m, col="red")
```



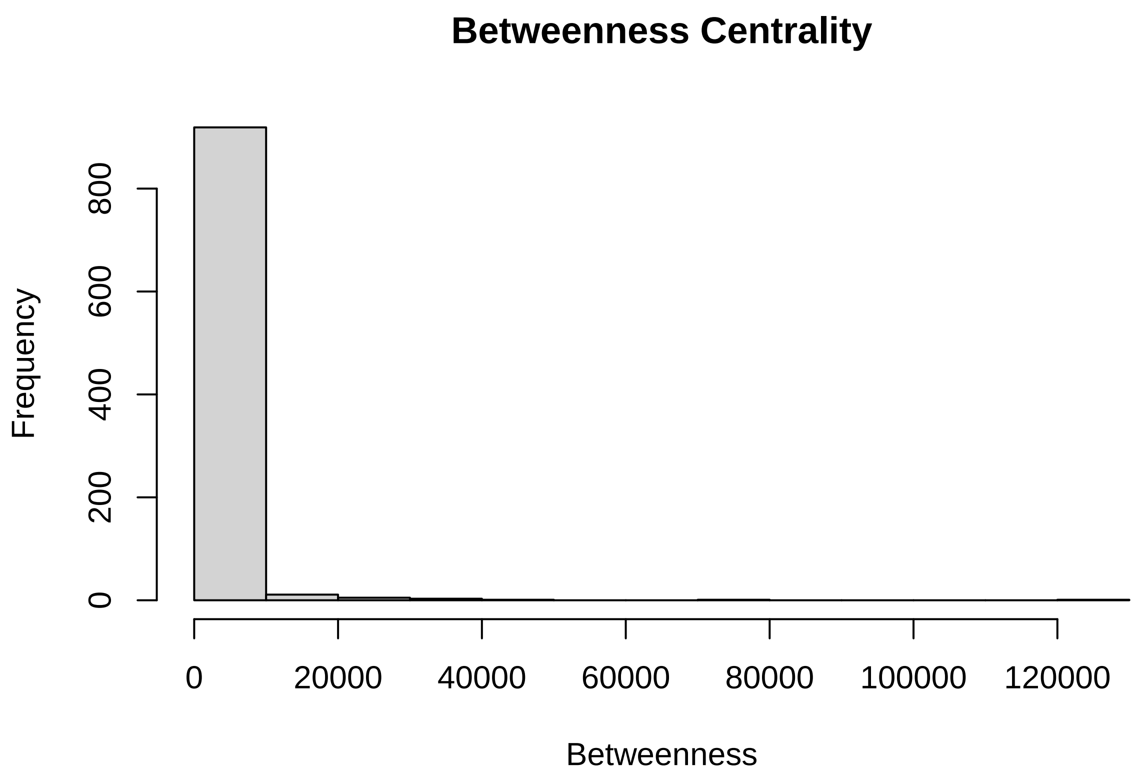
f -

What distributions do the following centrality measures follow:

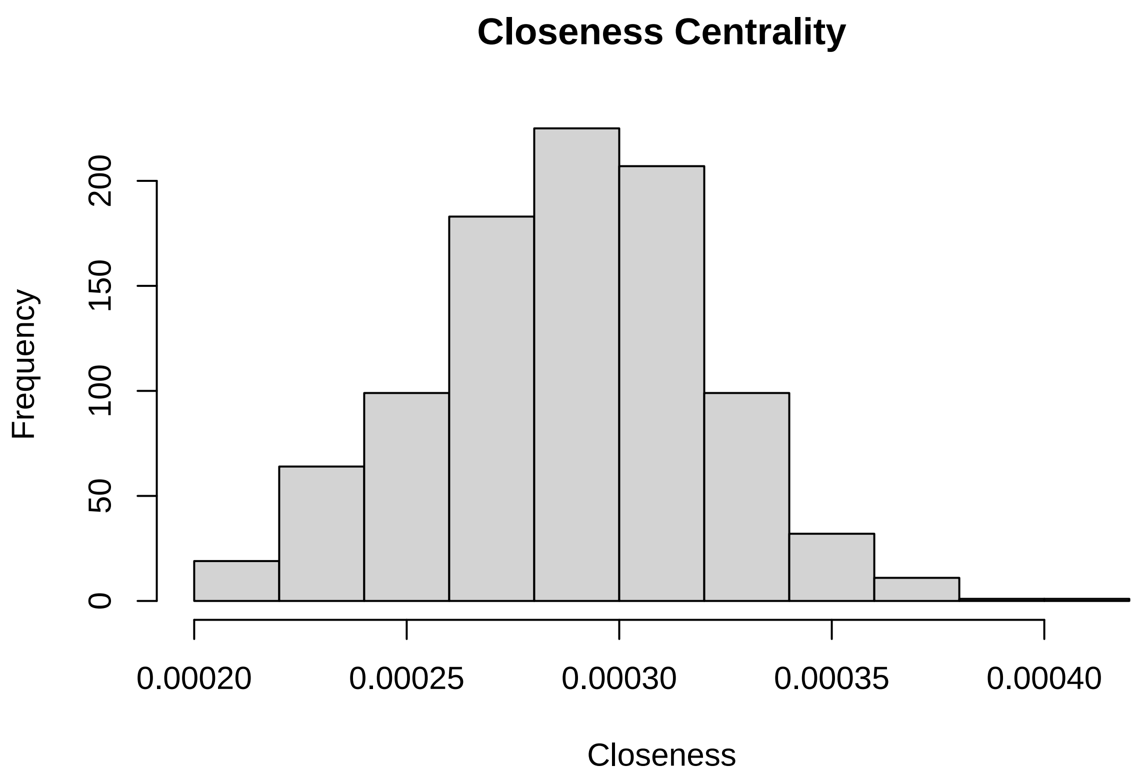
```
# Eigenvector
eigen_centrality <- eigen_centrality(H)$vector
hist(eigen_centrality, main="Eigenvector Centrality", xlab="Centrality")
```




```
# Betweenness
betw <- betweenness(H)
hist(betw, main="Betweenness Centrality", xlab="Betweenness")
```



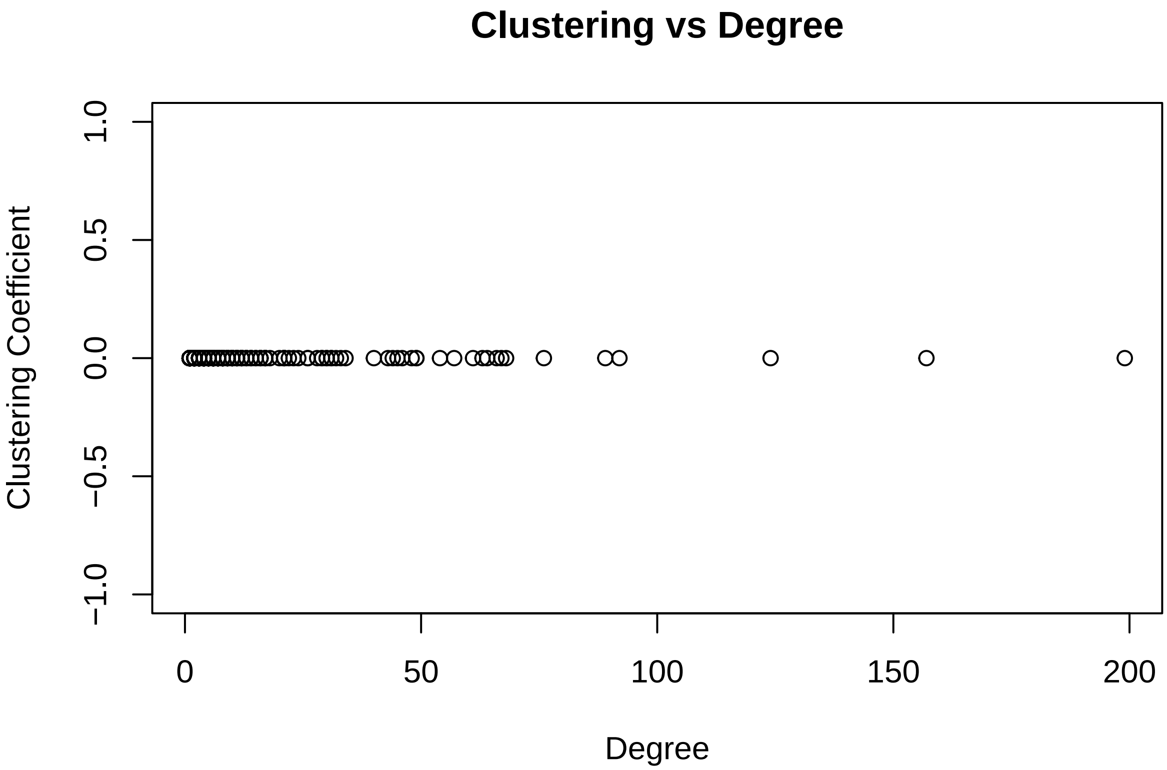

```r
# Closeness
closeness_centrality <- closeness(H)
hist(closeness_centrality, main="Closeness Centrality", xlab="Closeness")
```



g - How does the clustering coefficient of vertices change with vertex degrees?

FIX THIS

```
clustering <- transitivity(H, type = "local", isolates = "zero")
plot(deg, clustering, xlab="Degree", ylab="Clustering Coefficient", main="Clustering vs Degree")
```



h -

Does H exhibit assortative mixing in terms of vertex degrees? Provide the assortativity coefficient and interpret its value.

```
assortativity_degree <- assortativity_degree(H)
assortativity_degree
```

```
## [1] -0.3829303
```