

R Code: (If file doesn't save)

R version 4.4.1 (2024-06-14 ucrt) -- "Race for Your Life"
Copyright (C) 2024 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

```
> install.packages("igraph")
```

Installing package into 'C:/Users/soods/AppData/Local/R/win-library/4.4'

(as 'lib' is unspecified)

trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.4/igraph_2.1.1.zip'

Content type 'application/zip' length 7164520 bytes (6.8 MB)

downloaded 6.8 MB

package 'igraph' successfully unpacked and MD5 sums checked

The downloaded binary packages are in

C:\Users\soods\AppData\Local\Temp\RtmpOYY8tj\downloaded_packages

```
> library(igraph)
```

Attaching package: 'igraph'

The following objects are masked from 'package:stats':

decompose, spectrum

The following object is masked from 'package:base':

union

Warning message:

package 'igraph' was built under R version 4.4.2

```
> setwd("C:/Users/soods/OneDrive/Documents/COSC 421/Project")
```

```
> getwd()
```

```
[1] "C:/Users/soods/OneDrive/Documents/COSC 421/Project"
```

```
> nodes <- read.csv("Data_Nodes.csv", header = TRUE)
```

```
> edges <- read.csv("Data_Edges_2.csv" header = TRUE)
```

Error: unexpected symbol in "edges <- read.csv("Data_Edges_2.csv" header"

```
> edges <- read.csv("Data_Edges_2.csv", header = TRUE)
```

```
> g <- graph_from_data_frame(d = edges, vertices = nodes, directed = TRUE)
```

```
> betweenness centrality <- betweenness(g, directed = TRUE)
```

```
> eigenvector centrality <- eigen centrality(g, directed = TRUE)$vector
```

```
> top_5_betweenness <- sort(betweenness centrality, decreasing = TRUE)[1:5]
```

```
> top_5_eigenvector <- sort(eigenvector centrality, decreasing = TRUE)[1:5]
```

```
> avg_betweenness <- mean(betweenness centrality)
```

```
> avg_eigenvector <- mean(eigenvector centrality)
```

```
> plot(g, vertex.size = betweenness centrality * 10, main = "Betweenness Centrality")
```

```
> plot(g, vertex.size = eigenvector centrality * 10, main = "Eigenvector Centrality")
```

```
> print("Top 5 Betweenness Centrality:")
```

```
[1] "Top 5 Betweenness Centrality:"
```

```
> print(top_5_betweenness)
```

```
222 76 223 187 77
```

```
667.0667 666.7847 588.8121 584.0259 556.7886
```

```
>
```

```
> print("Average Betweenness Centrality:")
```

```
[1] "Average Betweenness Centrality:"
```

```
> print(avg_betweenness)
```

```
[1] 239.2013
```

```
> print("Top 5 Eigenvector Centrality:")
```

```
[1] "Top 5 Eigenvector Centrality:"
```

```
> print(top_5_eigenvector)
```

```
209 105 103 102 28
```

```
1.0000000 0.3100306 0.2940963 0.2765931 0.2720015
```

```
> print("Average Eigenvector Centrality:")
```

[1] "Average Eigenvector Centrality:"

> print(avg_eigenvector)

[1] 0.0187761

> print(betweenness_centrality)

215	109	63	12	60	187	285	138
303.437710	364.784164	247.608327	471.294880	139.451080	584.025946	81.397436	309.056679
82	191	78	5	4	265	1	52
256.464023	278.631048	219.968608	205.301894	244.450043	183.045714	254.637085	263.194989
179	244	50	13	206	205	2252	92
263.444143	338.597741	185.546982	393.007670	268.559449	349.503129	21.644970	139.583592
217	2156	310	166	34	204	298	642
224.994476	58.376266	183.159599	367.527837	298.706704	349.157898	216.374763	253.359776
300	122	115	196	172	140	57	6
359.174129	250.943772	62.562540	420.388313	360.211619	318.544570	301.891028	359.401100
218	73	14	89	130	32	464	99
357.002375	335.622780	197.266219	208.352169	174.778391	145.121875	329.069354	287.757178
41	152	129	193	28	21	74	565
254.738369	140.108936	299.549233	401.334534	229.862226	199.413260	442.089516	177.834723
480	58	273	79	221	256	2143	66
284.417542	217.075500	113.142777	228.578005	258.663065	187.500324	368.575091	335.106663
17	104	31	254	19	53	201	54
240.399122	202.055189	159.849697	261.594382	270.660610	369.017470	190.237926	355.073695
87	414	40	126	71	245	37	199
168.128393	17.997156	131.369103	326.274504	397.295288	144.853508	276.018041	342.759173
105	2131	444	124	2	415	30	313
109.441031	99.403423	353.793861	269.733966	178.829352	328.914008	265.915399	93.594200
98	55	242	83	88	176	225	190
211.290056	337.181333	152.400047	440.431059	196.105990	499.694142	276.903629	114.006651

144	174	65	44	147	282	154	45
383.525843	518.821715	309.494761	239.702794	369.233296	213.594122	180.870942	
110.892441							
484	197	123	47	143	186	77	10
324.798415	433.406021	204.925516	129.379452	290.786253	342.367814	556.788609	
301.219537							
532	139	112	8	75	257	716	2116
149.239575	345.269117	444.311173	445.222874	427.441126	140.832737	63.773054	
121.163941							
189	255	229	230	48	227	15	173
235.283984	224.642781	164.629902	177.689085	151.361135	234.075665	157.561934	
437.038467							
108	617	516	170	287	631	531	314
508.548295	221.855688	120.720073	493.591002	176.395293	249.651207	193.844883	
183.638729							
153	11	161	133	213	219	272	159
388.342713	376.450158	132.572104	204.936476	297.583481	309.141616	167.006541	
228.001314							
171	440	511	231	260	64	107	352
519.279996	168.798655	250.384063	151.321152	125.639510	290.679267	347.118276	
193.830989							
148	106	400	232	403	25	103	177
332.722718	158.309514	177.078557	199.370430	129.957692	391.394478	154.008221	
382.670920							
2114	2132	505	524	119	155	35	715
76.466577	71.052793	174.580276	109.455087	489.209716	245.351014	119.541801	91.083766
431	297	102	16	26	137	561	81
170.935972	167.883071	104.819763	308.027746	382.525596	163.531051	215.633096	
503.505667							
198	200	39	249	223	235	84	209
432.391888	70.898548	322.533886	88.686764	588.812101	310.825514	311.615525	
182.059158							
452	250	192	165	401	96	210	212
459.943027	239.372927	197.032159	239.647054	98.281397	181.793780	146.848290	
346.909272							
354	69	311	93	228	443	150	283
210.499513	375.120601	124.100874	101.475422	172.961705	196.623215	335.677529	
91.546608							
9	305	134	27	101	68	266	2157

```

160.894200 287.206766 378.292008 318.155634 96.347041 236.257458 165.530815
19.363532
    24    315    557    136    76    241    460    621
248.320795 422.594925 326.537865 414.767407 666.784710 126.566647 261.626768
48.069126
    554    72    211    281    253    236    611    278
219.790994 131.431553 285.780133 247.586689 220.191648 297.855178 63.431556
215.322161
    142    248    252    110    563    132    450    262
209.879905 280.195985 153.964474 394.449799 198.677726 295.154622 214.371858
355.634145
    222    202    239    207    482    258    261    280
667.066658 197.404372 153.330627 205.651730 167.600670 58.129200 140.088632
90.072640
    180    410    224    240    7    246    113    233
50.263361 159.399426 111.268266 208.739153 150.166566 195.318428 160.315150
52.951752
    997    308    526    114    981    980    995    991
20.988846 188.694270 86.017567 118.096634 36.489343 41.495997 5.842275 10.930642
    70    169    992    994    988    987    982    801
421.087865 16.778211 11.553652 1.103155 262.057246 22.519297 191.822088 5.811472
> print(eigenvector_centrality)
    215    109    63    12    60    187    285
3.857748e-03 2.867742e-03 9.212563e-03 1.353002e-02 3.112575e-02 2.631021e-02
7.568874e-04
    138    82    191    78    5    4    265
1.722607e-02 2.131204e-02 3.660115e-03 9.521478e-03 1.767598e-02 1.835655e-02
5.985045e-04
    1    52    179    244    50    13    206
3.065691e-03 8.882353e-03 5.894910e-04 1.051140e-03 1.265359e-02 7.477537e-03
2.168885e-03
    205    2252    92    217    2156    310    166
1.666829e-03 3.478069e-04 4.979252e-02 2.574272e-02 3.693404e-04 1.820928e-02
2.073227e-03
    34    204    298    642    300    122    115
2.470728e-02 1.017234e-02 8.954604e-04 3.234143e-03 2.271095e-02 4.843129e-02
4.780898e-04
    196    172    140    57    6    218    73
2.644384e-02 1.606148e-03 3.363487e-03 2.974708e-02 1.632112e-02 2.457368e-02
9.186389e-03

```

14	89	130	32	464	99	41
9.987636e-02	1.332879e-02	7.106063e-04	2.434375e-02	2.004308e-03	2.121479e-02	3.919089e-02
152	129	193	28	21	74	565
6.960119e-04	1.840334e-02	2.067164e-02	2.720015e-01	1.766761e-02	2.854871e-03	7.880939e-04
480	58	273	79	221	256	2143
1.081645e-03	1.589158e-02	2.718536e-04	1.540853e-02	3.200975e-03	3.861603e-04	4.852255e-04
66	17	104	31	254	19	53
1.788780e-02	3.866547e-02	2.493153e-01	1.821627e-01	8.924586e-04	1.293266e-02	2.335419e-02
201	54	87	414	40	126	71
4.452004e-02	3.975823e-03	8.642311e-03	4.400721e-04	2.597981e-02	8.853489e-03	2.160118e-03
245	37	199	105	2131	444	124
6.669943e-04	2.073155e-02	1.283600e-02	3.100306e-01	8.274128e-04	2.704636e-03	1.010725e-02
2	415	30	313	98	55	242
1.960354e-02	1.471907e-03	1.380235e-02	2.175773e-02	5.076098e-03	4.945152e-03	2.837302e-04
83	88	176	225	190	144	174
5.825665e-02	9.993172e-03	1.746026e-03	2.309966e-03	1.179475e-02	3.698479e-03	1.338217e-02
65	44	147	282	154	45	484
9.786812e-03	2.431397e-02	3.457037e-03	8.248088e-04	2.216238e-02	5.579120e-02	3.597335e-03
197	123	47	143	186	77	10
6.687253e-03	8.841906e-03	6.764937e-02	7.760920e-03	1.212454e-03	3.668785e-03	8.500169e-03
532	139	112	8	75	257	716
2.186777e-04	3.813894e-03	2.713527e-03	5.557125e-03	1.161465e-03	3.078325e-04	6.563487e-05
2116	189	255	229	230	48	227
8.295631e-04	2.659734e-02	1.332025e-03	3.605962e-03	4.000964e-03	1.137677e-02	4.881068e-03
15	173	108	617	516	170	287
7.877380e-03	1.028622e-02	1.048303e-02	1.718231e-03	7.830779e-05	4.467635e-03	2.280381e-03
631	531	314	153	11	161	133

1.595363e-03 7.103334e-04 1.823244e-02 2.345835e-03 3.144834e-02 2.800553e-03
1.077614e-03
213 219 272 159 171 440 511
3.301057e-03 3.518194e-03 1.297425e-03 6.479725e-03 6.364256e-03 1.602965e-03
6.993825e-04
231 260 64 107 352 148 106
1.102112e-03 6.806513e-04 6.491811e-03 4.655521e-03 2.182824e-04 1.695416e-02
9.039591e-04
400 232 403 25 103 177 2114
3.945511e-03 1.970176e-03 1.023876e-03 6.626537e-03 2.940963e-01 1.126452e-02
6.337757e-04
2132 505 524 119 155 35 715
6.162359e-04 1.160613e-03 2.805359e-04 7.440177e-03 6.485451e-03 2.976288e-02
2.619797e-04
431 297 102 16 26 137 561
9.108425e-04 3.403975e-03 2.765931e-01 1.371586e-02 9.008559e-03 2.226056e-02
2.410357e-03
81 198 200 39 249 223 235
3.292000e-03 4.228780e-03 9.054751e-02 1.466093e-02 2.483181e-04 1.641000e-03
1.388194e-03
84 209 452 250 192 165 401
4.316349e-02 1.000000e+00 2.552377e-03 1.434765e-03 3.491841e-03 2.371611e-03
5.366535e-04
96 210 212 354 69 311 93
1.543041e-02 8.660848e-04 8.249585e-03 3.208147e-03 3.420646e-03 1.079774e-03
7.656575e-02
228 443 150 283 9 305 134
3.213228e-03 1.200467e-03 4.345547e-03 3.345452e-04 4.700384e-03 7.171671e-03
1.338102e-03
27 101 68 266 2157 24 315
6.306828e-03 1.153280e-01 2.612505e-03 1.398109e-03 1.262376e-04 3.004502e-02
5.553630e-03
557 136 76 241 460 621 554
2.090325e-03 1.604425e-02 6.594524e-03 6.147686e-04 2.513960e-03 4.084378e-04
4.865880e-04
72 211 281 253 236 611 278
6.217763e-04 4.776180e-04 6.720003e-04 1.426844e-03 1.728360e-03 7.461279e-04
1.038781e-03
142 248 252 110 563 132 450

2.652235e-02 7.985064e-04 2.054773e-03 1.761840e-03 6.092091e-04 9.596313e-04
 1.712904e-03
 262 222 202 239 207 482 258
 1.036503e-03 4.369444e-03 6.679211e-04 5.518450e-04 1.335689e-02 9.845237e-04
 3.916411e-04
 261 280 180 410 224 240 7
 9.624753e-04 6.696046e-04 4.769550e-04 2.618949e-03 4.289904e-04 5.591533e-04
 4.086835e-04
 246 113 233 997 308 526 114
 5.791170e-04 5.244846e-04 5.161987e-04 2.603853e-04 7.157655e-03 2.491587e-04
 1.105356e-03
 981 980 995 991 70 169 992
 1.103418e-05 1.529491e-04 2.337603e-04 2.920090e-06 3.032067e-03 4.463073e-04
 9.262910e-05
 994 988 987 982 801
 4.140319e-04 1.893691e-05 7.154641e-06 2.763815e-05 1.206494e-04

```

> clusters <- cluster_louvain(g)
Error in cluster_louvain(g) :
  At vendor/cigraph/src/community/louvain.c:614 : Multi-level community
  detection works for undirected graphs only. Unimplemented function call
> clusters <- cluster_walktrap(g)
> V(g)$cluster <- membership(clusters)
> for (i in 1:max(V(g)$cluster)) {
+   subgraph <- induced_subgraph(g, V(g)$cluster == i)
+
+   subgraph_betweenness <- betweenness(subgraph, directed = TRUE)
+   plot(subgraph, vertex.size = subgraph_betweenness * 10, main =
+ paste("Betweenness Centrality - Cluster", i))
+   subgraph_eigenvector <- eigen_centrality(subgraph, directed = TRUE)$vector
+   plot(subgraph, vertex.size = subgraph_eigenvector * 10, main =
+ paste("Eigenvector Centrality - Cluster", i))
+ }
> save.image("~/COSC 421/Project/421 Project Code.RData")

```