Homework 8

Aaron Politsky

November 17, 2015

Part 1

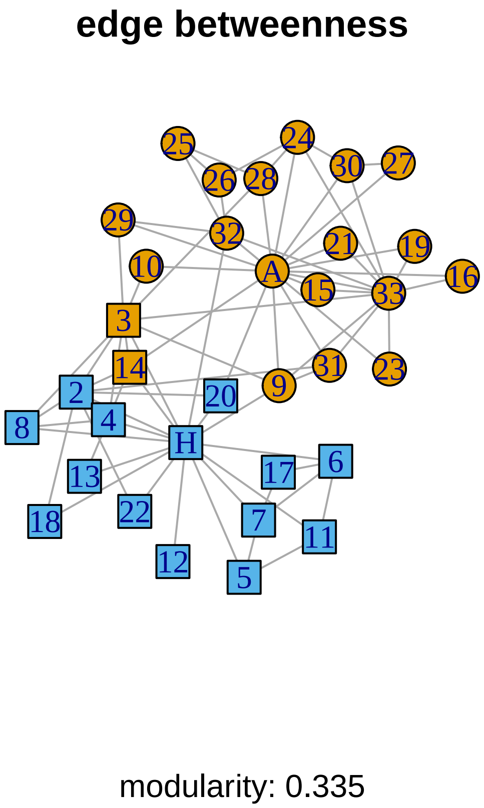
```
set.seed(99)
library(igraph)
library(igraphdata)
data(karate, package = "igraphdata")

# nodes in faction 1 will be rectangles
# nodes in faction 2 will be circles
shapes = c('rectangle', 'circle')
faction_vertex_shape = get.vertex.attribute(karate, "Faction")
faction_vertex_shape[faction_vertex_shape == 1] = shapes[1]
faction_vertex_shape[faction_vertex_shape == 2] = shapes[2]

# store layout so that it does not change for different plots
if (file.exists("karate.layout.Rdata")) {
  load("karate.layout.Rdata")
} else {
  karate_layout <- layout.davidson.harel(karate)
}
```

Generate communities using a variety of algorithms:

```
clustering.functions <- c(cluster_edge_betweenness,
                          cluster_fast_greedy,
                          cluster_infomap,
                          cluster_label_prop,
                          cluster_leading_eigen,
                          cluster_louvain,
                          cluster_optimal,
                          cluster_spinglass,
                          cluster_walktrap)
comms <- lapply(clustering.functions, function(cf) {cf(karate)})
```

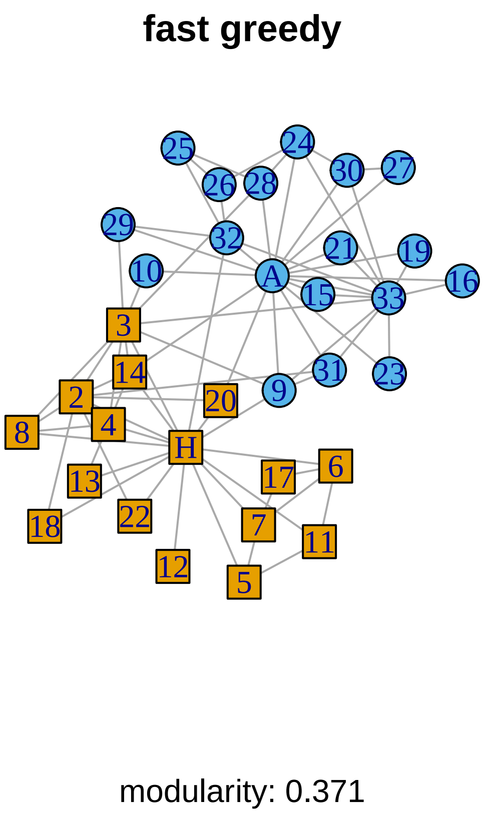
First, examine the hierarchical communities:

```
c <- comms[[1]]
cl2 <- cutat(c, no=2)
plot(karate,
     layout=karate_layout,
     vertex.shape=faction_vertex_shape,
     vertex.color=cl2
)
title(main=c$algorithm, sub=paste("modularity:",
                                  formatC(modularity(karate, cl2),digits = 3)))
```



Edge Betweeness is hierarchical and when cut to two communities, has only two error nodes.

```
c <- comms[[2]]
cl2 <- cutat(c, no=2)
plot(karate,
     layout=karate_layout,
     vertex.shape=faction_vertex_shape,
     vertex.color=cl2
)
title(main=c$algorithm, sub=paste("modularity:",
                                  formatC(modularity(karate, cl2),digits = 3)))
```

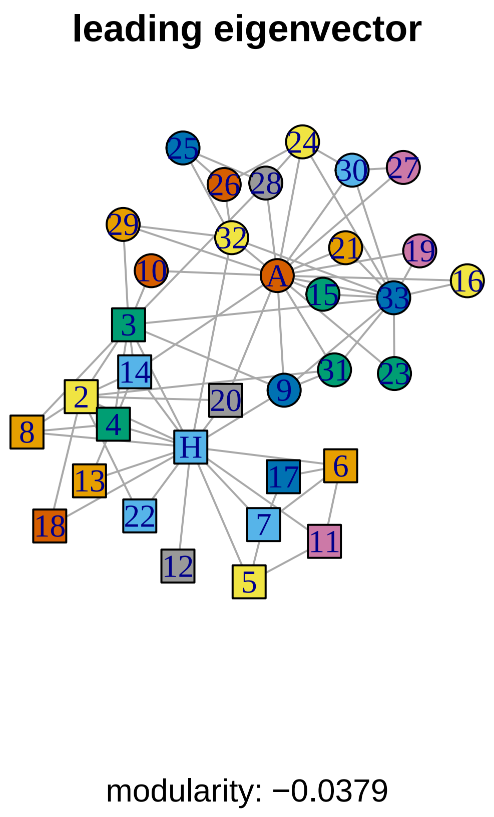


Fast Greedy actually gets it right. Impressive.

```
c <- comms[[5]]
cl2 <- cutat(c, no=2)
```

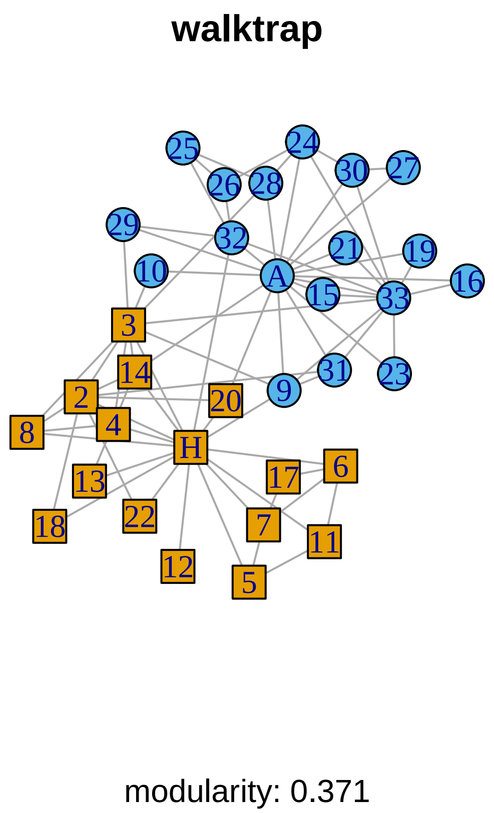
```
## Warning in cutat(c, no = 2): Cannot have that few communities
```

```
plot(karate,
     layout=karate_layout,
     vertex.shape=faction_vertex_shape,
     vertex.color=cl2
)
title(main=c$algorithm, sub=paste("modularity:",
                                  formatC(modularity(karate, cl2),digits = 3)))
```



Leading Eigenvector seems to want to use too many communities and complains about cutting.

```r
c <- comms[[9]]
cl2 <- cutat(c, no=2)
plot(karate,
     layout=karate_layout,
     vertex.shape=faction_vertex_shape,
     vertex.color=cl2
)
title(main=c$algorithm, sub=paste("modularity:",
                                  formatC(modularity(karate, cl2),digits = 3)))
```



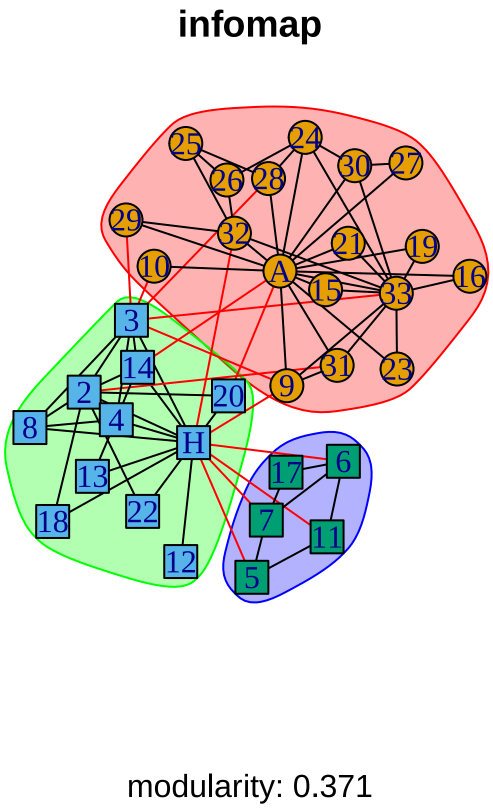
walktrap

modularity: 0.371

Walktrap actually gets it right, too.

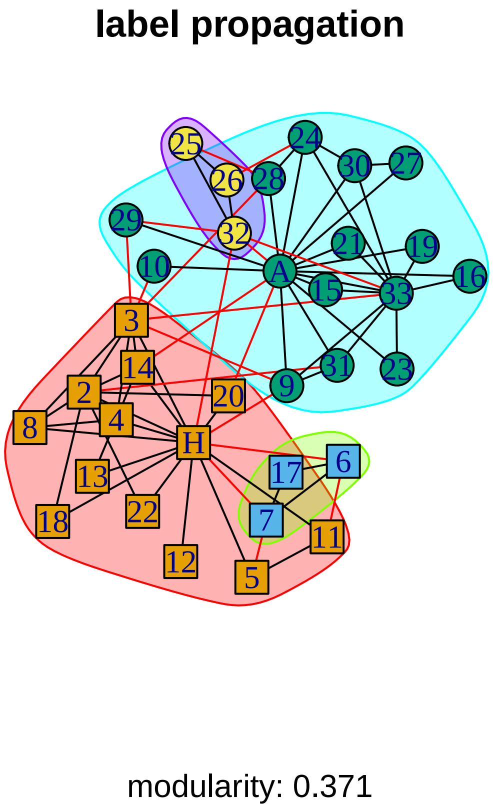
And on to the non-hierarchical communities:

```
c <- comms[[3]]
plot(x = c, karate, layout=karate_layout, vertex.shape=faction_vertex_shape)
title(main=c$algorithm, sub=paste("modularity:",
                                  formatC(modularity(karate, cl2),digits = 3)))
```



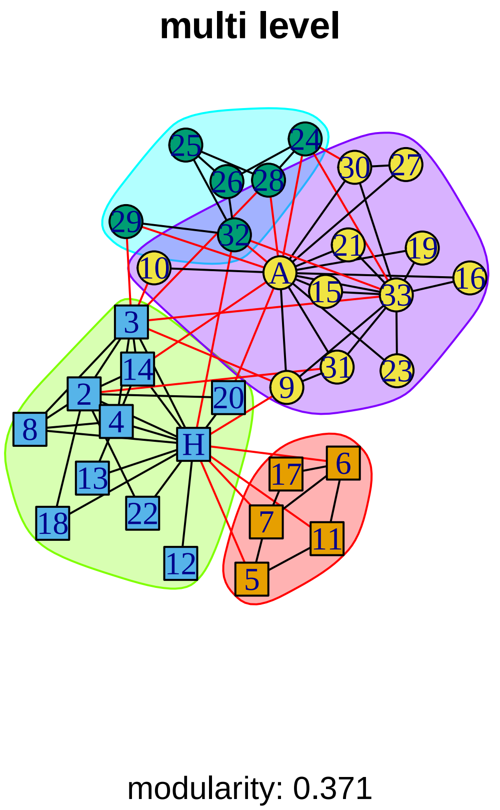
Infomap does pretty well, too.

```
c <- comms[[4]]
plot(x = c, karate, layout=karate_layout, vertex.shape=faction_vertex_shape)
title(main=c$algorithm, sub=paste("modularity:",
                                  formatC(modularity(karate, cl2),digits = 3)))
```



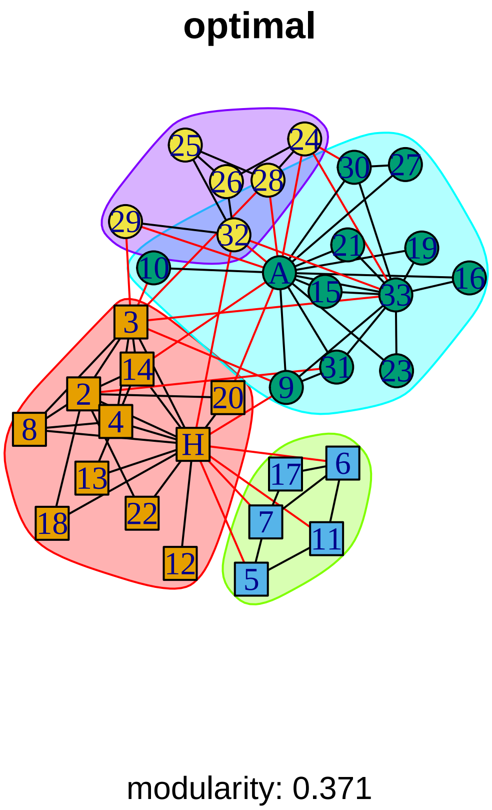
Label Prop splits nearly the same as Infomap, though it sub-factions the true A community.

```
c <- comms[[6]]
plot(x = c, karate, layout=karate_layout, vertex.shape=faction_vertex_shape)
title(main=c$algorithm, sub=paste("modularity:",
                                  formatC(modularity(karate, cl2),digits = 3)))
```



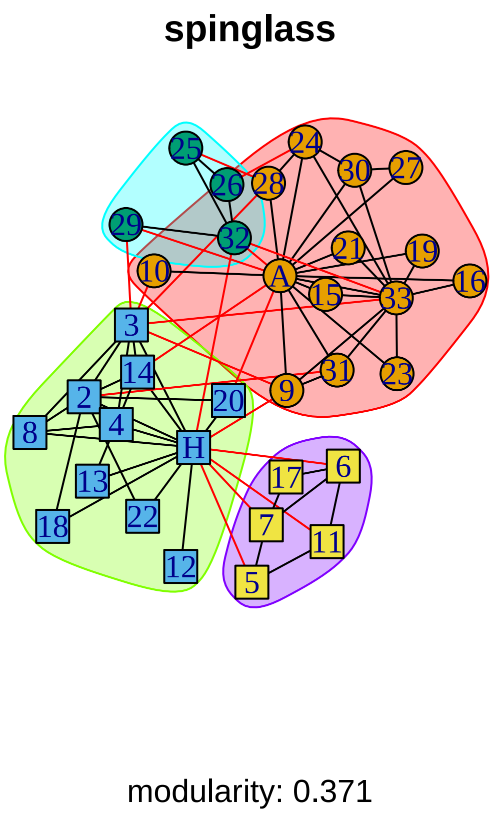
Multilevel wants to sub-faction the true A community.

```
c <- comms[[7]]
plot(x = c, karate, layout=karate_layout, vertex.shape=faction_vertex_shape)
title(main=c$algorithm, sub=paste("modularity:",
                                  formatC(modularity(karate, cl2),digits = 3)))
```



Optimal? Not so much. Same as Multilevel.

```
c <- comms[[8]]
plot(x = c, karate, layout=karate_layout, vertex.shape=faction_vertex_shape)
title(main=c$algorithm, sub=paste("modularity:",
                                   formatC(modularity(karate, cl2),digits = 3)))
```



Spinglass, similar to above.

Part 2

```
wg = read.graph("wikipedia.gml", "gml")
summary(wg)
```

```
## IGRAPH D--- 27475 85729 --
## + attr: id (v/n), wikiid (v/n), label (v/c)
```

```
list.vertex.attributes(wg)
```

```
## [1] "id"     "wikiid" "label"
```

```
get.vertex.attribute(wg, "label")[1:5]
```

```
## [1] "Dwarf corydoras"              "Homochronous"
## [3] "Telephony Application Server" "Life history"
## [5] "Wess-Zumino model"
```

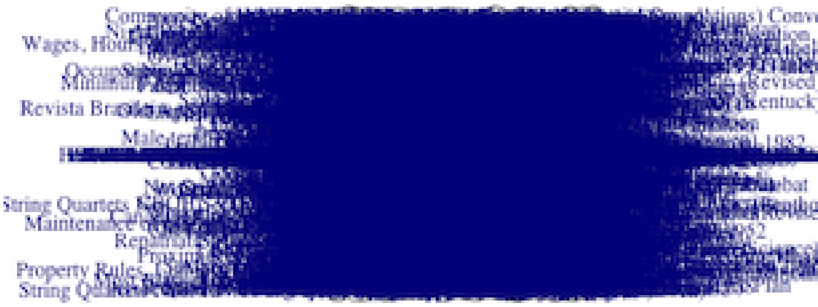
```
# some algorithms work on undirected graphs only
wgu = as.undirected(wg)
summary(wgu)
```

```
## IGRAPH U--- 27475 74148 --
## + attr: id (v/n), wikiid (v/n), label (v/c)
```

```
# compute communities and visually inspect if they make sense
```

```
system.time(lb <- cluster_label_prop(wgu))
system.time(fg <- cluster_fast_greedy(wgu))
system.time(lv <- cluster_louvain(wgu))
system.time(wt <- cluster_walktrap(wg))
system.time(im <- cluster_infomap(wg))
system.time(le <- cluster_leading_eigen(wg))
```

I'm omitting some methods which seem like they would take too long.



This graph seems to be cumbersome to visually inspect:

Analysis of vertex labels

As opposed to graphing.

Fast Greedy:

```
set.seed(99)
sample(get.vertex.attribute(wgu, "label")[membership(fg)==1], 15)
```

```
 [1] "Fibrillarin"                         "Sealant"
 [3] "Berthelot&apos;s reagent"            "Isoprostane"
 [5] "Metal-insulator transition"          "Resource energy"
 [7] "Englert-Greenberger duality relation" "Resistome"
 [9] "Periodic table (block)"              "Criegee rearrangement"
[11] "Hexafluoropropylene"                 "Sleeping pad"
[13] "Transition rule"                     "Batch reactor"
[15] "Cafergot"
```

Seems like the first group is scientific.

```
set.seed(99)
sample(get.vertex.attribute(wgu, "label")[membership(fg)==5], 15)
```

```
 [1] "Sufism"                     "Uniform Final Evaluation"   "Jobseeker&apos;s Allowance"
 [4] "Puppy love"                 "Populism"                   "Central Russian dialects"
 [7] "Gallup poll"                "Executive officer"          "General Secretary"
[10] "Major History of Islam"     "Chinese silver"             "Urban Gothic"
[13] "Gathering place"            "Ab-Zohr"                    "Byronic hero"
```

Some of these might be related.

```
set.seed(99)
sample(get.vertex.attribute(wgu, "label")[membership(fg)==25], 15)
```

```
 [1] "Corflu"
 [2] "Distinguish"
 [3] "Ad Astra (convention)"
 [4] "World Fantasy Convention"
 [5] "Case law"
 [6] "Acting president"
 [7] "Glamourcon"
 [8] "Majority opinion"
 [9] "Eurofurence"
[10] "List of Presidents of the American Bar Association"
[11] "MegaCon"
[12] "Marcon (convention)"
[13] "Michael S. Greco"
[14] "Legal case"
[15] "Foolscap"
```

Conventions?

```
set.seed(99)
sample(get.vertex.attribute(wgu, "label")[membership(fg)==50], 15)
```

```
 [1] "Worshipful Company of Coopers"
 [2] "Worshipful Company of Gardeners"
 [3] "Worshipful Company of Weavers"
 [4] "Mottos of the Livery Companies of the City of London"
 [5] "Worshipful Company of Girdlers"
 [6] "Worshipful Company of Cordwainers"
 [7] "Worshipful Company of Coachmakers and Coach Harness Makers"
 [8] "Worshipful Company of Dyers"
 [9] "Worshipful Company of Poulters"
[10] "Worshipful Company of Innholders"
[11] "Worshipful Company of Fletchers"
[12] "Worshipful Company of Tylers and Bricklayers"
[13] "Worshipful Company of Tin Plate Workers"
[14] "Worshipful Company of Plumbers"
[15] "Worshipful Company of Makers of Playing Cards"
```

Yep. Mostly.

```
set.seed(99)
sample(get.vertex.attribute(wgu, "label")[membership(fg)==100], 10)
```

```
[1] "List of volcanoes in the United Kingdom"
 [2] "List of volcanoes in Germany"
 [3] "List of volcanoes in Guadeloupe"
 [4] "List of volcanoes in Ascension Island"
 [5] "List of volcanoes in Netherlands Antilles"
 [6] "Lists of volcanoes"
 [7] "List of volcanoes in French Southern and Antarctic Lands"
 [8] "List of volcanoes in Wallis Islands"
 [9] "List of volcanoes in Western Samoa"
[10] "List of volcanoes in Martinique"
```

I'm convinced.

Let's see how others did:

Label Propagation:

```
set.seed(99)
sample(get.vertex.attribute(wgu, "label")[membership(lb)==2], 15)
```

```
 [1] "Telecommunications Industry Association"       "Leapfrogging"
 [3] "Telephone card"                                "Link protocol"
 [5] "Internet Radio Linking Project"                "The Strangest Secret"
 [7] "Telenet"                                       "Worldwide Digital Cordless Telecommunications"
 [9] "Call set-up time"                              "Malicious Caller Identification"
[11] "Fluency Voice Technology"                      "Synchronizer"
[13] "Foreign exchange office"                       "Digital communications"
[15] "Out-of-band"
```

These look to be telecom related.

```
set.seed(99)
sample(get.vertex.attribute(wgu, "label")[membership(lb)==12], 15)
```

```
 [1] "Mylogon"                      "GoToMyPC Pro"                 "PrintableString"
 [4] "LMHOSTS"                      "XHTML Mobile Profile"         "COMPEQ"
 [7] "EXeem"                        "NetBIOS"                      "TeamSpeex"
[10] "Virtual microscope"           "Stub (distributed computing)" "Port knocking"
[13] "TCP/IP stack fingerprinting"  "Reverse proxy"                "Computer data logging"
```

computer networking.

Louvain:

```r
set.seed(99)
sample(get.vertex.attribute(wgu, "label")[membership(lv)==678], 15)
```

```
 [1] "Canonical form (Boolean algebra)" "Inequation"
 [3] "Tsirelson space"                  "Spence&apos;s function"
 [5] "Scott information system"         "Quasi-finite morphism"
 [7] "List of mathematicians (V)"       "Algebraic graph theory"
 [9] "List of mathematicians (D)"       "Nowhere-zero flows"
[11] "Hereditary ring"                  "Fisher&apos;s z-distribution"
[13] "K-finite"                         "Hilbert&apos;s sixth problem"
[15] "Subsequential limit"
```

Mathy

```r
set.seed(99)
sample(get.vertex.attribute(wgu, "label")[membership(lv)==259], 15)
```

```
 [1] "Sensory Logical Extrovert"   "Champion (role variant)"     "Ethical Sensory Extrovert"
 [4] "Counselor (Role Variant)"    "Ethical Sensory Introvert"   "Healer (Role Variant)"
 [7] "Sensory Ethical Introvert"   "Protector (role variant)"    "Mastermind (Role Variant)"
[10] "True Colors Inc."            "Provider (role variant)"     "Intuitive Logical Extrovert"
[13] "Intuitive Ethical Introvert" "Ethical Intuitive Introvert" "Logical Sensory Introvert"
```

Personality?

```r
set.seed(99)
sample(get.vertex.attribute(wgu, "label")[membership(lv)==803], 15)
```

```
 [1] "Quiz Call"                                    "Sandra Corleone"
 [3] "Chinese art"                                  "Design methods"
 [5] "List of New York City housing cooperatives"    "Garden designer"
 [7] "Confucius Plaza"                              "Paulie Fortunato"
 [9] "Design firm"                                  "Design"
[11] "Society of London Art Dealers"                "Timeline of the War in Afghanistan (May 2002)"
[13] "Momo Barone"                                  "Inverter (air conditioning)"
[15] "Blade"
```

Not so sure about this one.

Infomap

```r
set.seed(99)
sample(get.vertex.attribute(wgu, "label")[membership(im)==1], 15)
```

```
 [1] "Ring homomorphism"                    "Rule of nines (mathematics)"
 [3] "Wheel theory"                         "Multiplicative distance"
 [5] "Primitive ring"                       "Ultrapower"
 [7] "Philosophy of mathematics education"  "Algebraic element"
 [9] "Associative algebra"                     "Example of a non-associative algebra"
[11] "Splitting field"                         "Macaulay computer algebra system"
[13] "Graded vector space"                     "Ordered field"
[15] "Moore method"
```

Mathy.

```
set.seed(99)
sample(get.vertex.attribute(wgu, "label")[membership(im)==387], 15)
```

```
 [1] "Angelic Society"                       "Egbo"
 [3] "Power behind the throne"               "Secret society"
 [5] "Bow Down (Mind of Mencia)"             "Skull and Dagger"
 [7] "Abraham ben Levi Conque"               "Camarilla (history)"
 [9] "Lineage-bonded society"                "Leopard Society"
[11] "Secrecy (sociology)"                   "Thomasine Church (Gnostic)"
[13] "Religion in Nigeria"                   "Village-bonded society"
[15] "Conspiracy theory (disambiguation)"
```

Secret societies and Carlos Mencia. Carlos Mencia runs a secret society!

Well, I could go on, but I can see these are pretty effective from just crunching on the graphs.